
Sistemas de Visión Frameless, Procesado Hardware AER y Procesado de la Información Mediante Eventos

Septiembre 2018



VNIVERSITAT
DE VALÈNCIA

Autor: D. Juan Barrios Avilés

Director: D. Alfredo Rosado Muñoz

Departamento de Ingeniería Electrónica
UNIVERSITAT DE VALÈNCIA

Agradecimientos

La realización de esta tesis ha sido posible a la ayuda de mucha gente que directa o indirectamente me han apoyado.

Primeramente, agradezco al CONACYT por el soporte brindado durante la mayor parte de la realización de este trabajo.

Agradezco también a todos los compañeros de laboratorio que he tenido a lo largo de los estudios de doctorado, que con sus consejos, opiniones y colaboraciones he podido mejorar y complementar mi tesis. De igual manera, agradezco a los profesores del máster DIMSAI y a la delegación de B&R Valencia, que con su experiencia me asesoró durante la ejecución de las distintas pruebas que se presentan en este trabajo.

A mi director de tesis, Alfredo, por su ayuda, paciencia y dedicación que me facilitaron encontrar la orientación de mis estudios y de la culminación de esta tesis.

A Carla por el apoyo y fe en mí a lo largo de todos estos años.

Y finalmente a mis padres y hermanos por haberme fomentado los deseos de conocimiento durante toda mi etapa formativa, además de su apoyo incondicional.

A todos Gracias.

Resumen

Estimado lector, seguramente el día de hoy te has cruzado en el campo de acción de algún equipo de visión y posiblemente no te has dado cuenta, al utilizar el transporte público, al sacar dinero del cajero automático, al hacer la compra en el supermercado o al tomar una foto con tu teléfono móvil. La democratización de las cámaras de visión en nuestra vida cotidiana ha tomado fuerza pues cada vez son más pequeñas, con procesamiento más potente y menor consumo de energía, facilitando su implementación en diferentes dispositivos de tamaños que hasta hace pocos años se creía imposible y que ahora encontramos en casi cualquier lugar.

En ocasiones, la motivación de los distintos fabricantes de estos dispositivos en competir por el mercado ha incrementado las capacidades de sus productos proporcionándoles características que ya no es posible distinguir a simple vista si no se tiene una aplicación en concreto. El ejemplo más representativo es el sobrepasar resolución de calidad de imagen más allá de lo que el ojo humano es capaz de distinguir. No obstante, existen aplicaciones de visión artificial en las que el desempeño apenas se ha visto mejorado en las últimas décadas. Se trata de

aquellas en las que prima la velocidad de procesamiento más allá de la resolución y calidad de la imagen. Estas aplicaciones las abarca principalmente el sector de la industria de la manufactura y procesos, con aplicaciones de lectura de código de barras, inspección de calidad, inspección de montaje, guiado y posicionamiento de robots entre muchas otras que se derivan de estas. Aunque la visión artificial en el sector de la automatización industrial ya comienza a ser una tecnología madura y bien aceptada, existen diversas limitaciones en los sistemas convencionales de captura de imagen tales como la variabilidad de la iluminación y la captura de información redundante por citar las principales, que repercute en una mayor cantidad de datos a almacenar y procesar, además de poca flexibilidad a cambios de la iluminación en la escena no planificados. Todo ello supone que los sistemas de visión artificial industrial sean aún un campo por explorar y mejorar.

Los sistemas de visión basados en fotogramas o *frames* son los sistemas convencionales que se montan en casi cualquier cámara, ya sea en la industria, en nuestro portal de la casa o en nuestros ordenadores. Estos sistemas distan mucho de la forma en la que los organismos biológicos con ojos ven, pues los primeros necesitan de un disparo para la captura de una matriz de píxeles, mientras que los segundos capturan la información de manera continua sin necesidad de ningún disparo, es decir, sin tener que crear una matriz de píxeles a cada instante, creando un flujo continuo de información.

En los sistemas de visión basados en fotogramas, los píxeles son generados, almacenados y procesados sin necesariamente tener alguna información nueva de la escena, por lo que son datos que posiblemente ya se han procesado y que no aportan nada, desperdiciando de esta manera energía en el procesamiento y capacidad de almacenamiento. En los organismos biológicos, la capa del ojo que recibe los estímulos de las diferentes ondas de luz y en donde se imprime la escena, llamada retina, no precisa de un disparador de captura pues procesa de manera continua conforme se van estimulando las células fotorreceptoras que

posee a símil de píxel. Estos estímulos son llevados en forma de impulsos por un único conducto al cerebro codificados en tiempo y espacio para su procesamiento y toma de decisiones. El tiempo de transmisión y procesamiento es tan pequeño que para movimientos mecánicos que pueden realizar los organismos es despreciable.

Por otra parte, la tendencia de todos los circuitos electrónicos integrados a ser más pequeños y con mayores capacidades han facilitado que los píxeles también se vean beneficiados de estos avances. Con esto se han podido crear sistemas que hace 30 años se tenían ya en mente pero que la tecnología del momento no permitía hasta hace apenas algunos años, como lo son las cámaras basadas en eventos, que tienen un enfoque bio-inspirado neuromórfico. Esta tecnología de cámaras es más semejante a cómo funciona el ojo, produciendo estímulos, impulsos o eventos según los cambios de intensidad de iluminación, los cuales son transmitidos a una unidad de procesamiento utilizando generalmente una codificación para la representación de eventos. Una de las codificaciones mayormente empleadas es la Representación de Eventos por Direccionamiento o *AER*. Este es un método de codificación eficiente y universal para transmitir datos en formato de evento. Con este tipo de codificación, cada dispositivo solo transmite información en caso de cambios de estado en sus sensores. Lo que trae consigo otras necesidades para una correcta codificación y transmisión de la información: el uso de tecnologías específicas para la comunicación.

En la actualidad, los sistemas de visión artificial basados en fotogramas usualmente requieren de canales de comunicación dedicados y cerrados para transmitir los datos a la unidad de procesamiento, haciendo muy difícil converger con canales de comunicación que transmiten información de otros dispositivos. A pesar de que en los últimos años existe una tendencia a unificar la mayoría de los protocolos, en los sistemas que incorporan visión artificial no existe unanimidad ni en la capa física a utilizar como puede ser USB o Ethernet y tampoco en el protocolo a incorporar. De manera similar, en la industria de ma-

nufactura y procesos esto pasaba a finales del siglo pasado en donde ni las capas físicas ni los protocolos convergían para la transmisión de información entre los distintos equipos. Sin embargo, desde hace aproximadamente una década el mercado ha optado cada vez más por unificar los protocolos en una misma capa física y la elegida ha sido Ethernet. La elección de esta capa física de transporte de información también ha contribuido a que el abanico de protocolos de comunicación se reduzca con la incorporación de los distintos fabricantes de equipos industriales a protocolos existentes ya probados e implementados en diversos sectores.

Esta tesis describe el trabajo realizado para aprovechar características de dispositivos generadores de eventos, con especial énfasis en los dispositivos de visión, en un sector poco explorado como es el industrial tratando su procesamiento y transmisión. Para lograr esto se pasa desde convertir sensores convencionales a generadores de eventos, el procesamiento de la información en eventos con principal interés en las necesidades que el sector industrial requiere, hasta la implementación de un sistema generador de eventos en un sistema de automatización y su comparación con uno convencional. Para la presentación de algoritmos, codificaciones, pruebas y resultados se ha dividido este escrito en los siguientes capítulos:

Capítulo 1: Introducción

En el primer capítulo se exponen los motivos por lo que es interesante explorar sistemas como los generadores de eventos en diferentes sectores donde sus cualidades son importantes y que han sido poco exploradas hasta la fecha y que pueden causar un impacto positivo al desarrollo del sector. Se establecen los objetivos de la investigación y los alcances que puede lograr este trabajo en el desarrollo de tecnologías basadas en eventos en un sector poco explorado por dispositivos bio-inspirados como el de la automatización industrial.

Capítulo 2: Marco Teórico

En este capítulo se hace un repaso teórico de los principales ejes donde se ha desarrollado la investigación: sistemas de visión bio-inspirados, su procesamiento y transmisión la información y la incorporación de estos a la industria de procesos y manufactura. Con el objetivo de poder incorporar un sistema generador de eventos bio-inspirado en una red industrial, y concretamente de un sistema de visión, se hace un recorrido de la evolución de los sistemas de visión orgánicos, su funcionamiento, y de la necesidad de los investigadores por crear sistemas bio-inspirados neuromórficos. Esto se enlaza con los métodos de procesamiento habituales que utilizan los sistemas bio-inspirados generadores de eventos como las Redes Neuronales Pulsantes (SNN), o las Redes Neuronales Convolucionales (CNN). Se explica el principal método de codificación y transmisión de eventos generados por estos dispositivos llamado AER. Finalmente se establece el panorama, tendencias y estándares que existen en la industria para poder intercambiar datos entre equipos. Este intercambio se puede hacer de diferentes maneras tanto en el medio físico que transporta la información como en la manera de codificarla. Siempre considerando, que este intercambio de información se verá acotado por la exigencia y necesidades de la aplicación.

Capítulo 3: Codificación en Eventos para Sensores no Basados en Eventos

En este capítulo se aborda la posibilidad de poder convertir dispositivos convencionales en dispositivos generadores de eventos, considerando que esta información podrá ser tratada por otros sistemas o dispositivos que reciben los datos en forma de eventos. De esta manera ventajas como la eliminación de datos redundantes, y la reducción del tamaño de almacenamiento pueden trasladarse a sistemas convencionales. Ampliando el abanico de dispositivos disponibles generadores de eventos que puedan seleccionados para aplicaciones fuera del ám-

bito de la investigación. Desde luego, esta conversión debe realizarse de manera sencilla y a bajo coste, motivo por el cual en este capítulo se proponen conversiones sencillas y con mínimas modificaciones. Se han hecho pruebas con tres sensores utilizados normalmente en la robótica móvil: sensor de ultrasonidos, compás digital y cámara de visión. Los resultados muestran que con la implementación y codificación propuesta se puede obtener información análoga a la que produce un sensor bio-inspirado generador de eventos.

Capítulo 4: Algoritmos de Procesado de Eventos

A pesar de que un sistema de visión basado en eventos genera menor cantidad de datos que un sistema convencional de visión basado en fotogramas, la cantidad de datos puede resultar aún alta para transferir especialmente en ambientes donde conviven diversos dispositivos que intercambian información de manera cíclica en una misma red. En este capítulo se presenta el algoritmo LDSI, el cual reduce la cantidad de datos manteniendo la misma información. Con este algoritmo se acondicionan los datos para poder ser transmitidos en una red de comunicaciones sin saturarla, además de no tener que crear una línea exclusiva para dicha transmisión. El algoritmo ha sido diseñado parametrizable para lograr diferentes niveles de reducción de datos. La sintonización de los parámetros del algoritmo para lograr una reducción es tarea difícil al tener un comportamiento no lineal. Para resolver este problema, se plantean dos algoritmos, ESM y ENM, que operan en conjunto con la finalidad de encontrar dos indicadores numéricos que ayuden a encontrar el nivel de reducción y la similitud entre dos conjuntos de eventos y así comparar resultados obtenidos del algoritmo LDSI. Con la propuesta de los algoritmos de este capítulo es posible preparar sistemas de visión basados en eventos de manera sencilla para su incorporación en una red industrial.

Capítulo 5: Análisis de Protocolos de Comunicación Basados en Ethernet para Transmisión de Eventos

El panorama de posibilidades para incorporar nuevos dispositivos dentro de una red industrial es amplio. Es muy diversos desde la manera de codificar la información, el método de transferencia y el medio por donde se envían los datos. Desde hace poco más de una década la tendencia de los distintos fabricantes de equipos industriales es a utilizar Ethernet como medio de transferencia, no obstante, no ha habido consenso en la forma de codificar ni en la metodología del envío. En este capítulo se ponen a prueba dos de los protocolos de comunicación industrial basados en Ethernet que poseen características interesantes para la incorporación de los datos generados en forma de eventos. Los protocolos elegidos son Ethernet/IP y POWERLINK, ambos con un fuerte respaldo en aplicaciones demandantes en la velocidad de transmisión de información como es el sector del control de movimiento o control de ejes. Se presentan diferentes pruebas para designar cuál de los dos protocolos proporciona un mejor desempeño con la finalidad de incorporar los datos generados por dispositivos generadores de eventos, concretamente en este caso, de un sistema de visión.

Capítulo 6: Sistema de Visión Basado en Eventos en la Automatización Industrial

Ya definido el medio y forma de transmisión y acondicionados los datos por medio del algoritmo LDSI, se desarrolló una plataforma a modo de proceso industrial en donde se pudiera incorporar un sistema de visión basado en eventos y ponerlo a prueba bajo condiciones demandantes como es el seguimiento de un objeto a alta velocidad. Además, se incorporó un sistema convencional de visión artificial basado en fotogramas que nos pudiera arrojar datos comparativos entre el desempeño de ambos sistemas. Los resultados destacan las ventajas en cuanto a velocidad de respuesta y poca sensibilidad que poseen los sistemas de visión basados en eventos, dos de los factores más determinantes a

la hora de diseñar un proceso industrial que incorpora visión artificial en sus tareas. Los resultados obtenidos en este capítulo dieron la oportunidad de participar en el concurso ETHERNET-POWERLINK AWARD 2016/2017 [1]. El trabajo fue seleccionado de entre veintidós universidades participantes y gracias a las novedades que incorpora el sistema y a la relevancia del mismo en el sector resulto ganador con el primer lugar.

Capítulo 7: Conclusiones

En el último capítulo se presentan las conclusiones de los resultados obtenidos, en base a los objetivos planteados al inicio. Se hace un análisis de los resultados al convertir sensores convencionales a sensores generadores de eventos, el procesamiento de los mismos, el acondicionamiento de la información según las necesidades de la aplicación final y se hace un resumen de las características que un sistema de visión basado en eventos puede aportar a un campo poco explorado por este tipo de tecnologías como es el de la industria de procesos y fabricación.

Juan Barrios Avilés

juanba7@uv.es

Índice general

Agradecimientos	III
Resumen	V
Índice general	XIII
1 Introducción	1
1.1 Contexto General	1
1.2 Motivación	3
1.3 Objetivos	5
2 Marco Teórico	7
2.1 La Vista y su Evolución	8
2.2 Funcionamiento del Ojo	11

2.3	La Neurona	14
2.4	Sistemas Neuromórficos	17
2.5	Representación de Eventos por Direccionamiento, AER	21
2.6	Procesamiento de información mediante eventos	24
2.7	Comunicaciones Industriales	25
2.7.1	Comunicaciones Industriales en la Actualidad	26
2.7.2	Características de las Comunicaciones Industriales	28
2.7.3	Redes Industriales Ethernet	29
2.7.4	Principales Protocolos de Comunicación basados en Ethernet	31
3	Codificación en Eventos para Sensores no Basados en Eventos	39
3.1	Codificación en Eventos	40
3.2	Generación de Eventos y Codificación AER para Sensores de Ultrasonidos y Compás Digital	42
3.2.1	Tipos de Generación de Eventos.	42
3.3	Generación de Eventos y Codificación AER para Cámara Basada en Fotogramas	43
3.4	Pruebas	46
3.4.1	Sensor de Ultrasonidos	46
3.4.2	Compás Digital	51
3.4.3	Cámara Basada en Fotogramas	54
3.5	Resumen	60
4	Algoritmos de Procesado de Eventos	63
4.1	Punto de Partida	65
4.2	Algoritmo LDSI	66
4.2.1	Verificación del Algoritmo LDSI.	72
4.2.2	Descripción del Hardware	74

4.2.3 Resultados del Algoritmo LDSI	76
4.3 Metodología de comparación	87
4.4 Algoritmo de Medición del Ruido en Eventos, ENM	89
4.5 Algoritmo de Comparación de Bordes, ESM.	93
4.6 Resultados de los algoritmos ENM y ESM	94
4.7 Resumen	98
5 Análisis de Protocolos de Comunicación Basados en Ethernet para Transmisión de Eventos	101
5.1 Panorama en las comunicaciones industriales	103
5.2 Ethernet/IP y POWERLINK	105
5.3 Pruebas comparativas entre Ethernet/IP y POWERLINK	106
5.4 Resultados	109
5.4.1 Análisis del tiempo de respuesta mediante la transferencia de la cantidad mínima de datos	109
5.4.2 Análisis del tiempo de respuesta mediante la transferencia de la cantidad máxima de datos.	112
5.4.3 Análisis de la compatibilidad de la red industrial convergente con una red corporativa convencional.	114
5.4.4 Análisis de la resistencia a los ciberataques	114
5.4.5 Análisis de los tiempos de recuperación después de un fallo de comuni- cación	115
5.5 Resumen	116
6 Sistema de Visión Basado en Eventos en la Automatización Industrial	121
6.1 Generalidades	122

6.2 Configuración Hardware	124
6.2.1 Iluminación	135
6.3 Mecanismo Robótico	137
6.4 Pruebas	139
6.4.1 Procesamiento basado en eventos	139
6.4.2 Procesamiento basado en fotogramas	141
6.5 Puesta en marcha	142
6.6 Resultados	144
6.7 Resumen	163
7 Conclusiones	167
7.1 Codificación y conversión de eventos	167
7.2 Filtrado y tratamiento de eventos	169
7.3 Sistemas de eventos en la Automatización Industrial	171
7.4 Conclusiones Generales	173
Bibliografía	175

Capítulo 1

Introducción

¿Cuál es la motivación y propósito de este trabajo? ¿De dónde parte y hacia a dónde se encamina el cometido de esta tesis? Estas son algunas de las preguntas que se responden en este capítulo, en el que también se sitúa al lector en un panorama general de cualidades y características de los sistemas neuromórficos generadores de eventos. Estas pueden aprovecharse en campos en los que convencionalmente estas tecnologías no han tenido cabida, estableciendo al mismo tiempo los alcances esperados en los objetivos de la tesis.

1.1 Contexto General

Desde la aparición de los sistemas computacionales, el costo del procesamiento ha estado directamente relacionado a su gasto energético. Ya en la década de los años cuarenta del siglo pasado con la construcción de los primeros ordenadores de propósito general como el ENIAC [2], [3], se consideraba el gasto energético de su operación en el que no solo preocupaba el gasto de energía computacional por si mismo sino la

energía total consumida también por los conectores, las tarjetas electrónicas, los ventiladores, las salidas de visualización y todos aquellos equipos que hacían posible el funcionamiento. Ejemplos como este han derivado en que los desarrollos tecnológicos en el sector hayan tenido un marcado enfoque en la mejora de la ratio de la energía consumida por unidad de procesamiento computacional y la eficiencia en materia de procesamiento.

Históricamente los sistemas biológicos han sido inspiración para la ingeniería [4], [5], [6], [7], [8], siendo determinantes tanto en la creación de dispositivos como en el desarrollo de algoritmos que emulen algún comportamiento biológico. La principal línea de investigación en el sector es la relacionada a la capacidad de los sistemas biológicos en cuanto a eficiencia energética por unidad de procesamiento, especialmente en el campo de desarrollo de nuevo hardware computacional. Para construir un sistema computacional bio-inspirado se deben encontrar las similitudes entre el sistema biológico a emular o en el que se basa y el sistema electrónico computacional, por ejemplo en un sistema nervioso existe una barrera energética construida por la diferencia en la constante dieléctrica entre distintas sustancias proteínicas, líquidas y lípidos, en la electrónica esta barrera es construida por la diferencia entre el espacio que hay entre los diferentes capas de silicio de cada transistor. Los enfoques de aplicaciones y procesamiento y transmisión de información a través de estos desarrollos bio-inspirados poco a poco tienen más cabida en distintos ámbitos fuera de los laboratorios de investigación.

En esta tesis se presenta una metodología de transmisión, procesamiento y filtrado de información generada por sistemas bio-inspirados y neuromórficos, aprovechando sus características para ser empleadas en campos poco explorados como es el sector industrial.

1.2 Motivación

En la actualidad, cada vez es mayor la demanda social por una respuesta directa del mundo tecnológico. La democratización del uso de los sistemas informáticos se ha visto incrementada en los últimos veinte años, gracias a su incorporación en el contexto de la vida cotidiana. Este avance global en la implantación de las tecnologías no ha presentado sino una mayor dependencia de la sociedad. La demanda de tecnología avanzada se ha incrementado exponencialmente, pasando de ser principalmente utilizada en la industria a representar una necesidad en la vida cotidiana. En esta misma línea, los sistemas de visión artificial se han incorporado paulatinamente en nuestras vidas como respuesta a la demanda que diversos sectores requieren como el productivo, la investigación y el ocio. Su desarrollo va de la mano tanto en los avances tecnológicos en hardware como software con marcadas tendencias de implementación a la miniaturización y a la eficiencia de uso de recursos.

Uno de los sectores donde mayor desarrollo ha habido es el relacionado a la visión artificial. Sin embargo, a pesar de que los sistemas de visión convencional llevan varias décadas evolucionando, han mantenido la característica de ejecutar un gran número de operaciones y procesar una gran cantidad de datos que no poseen información relevante para el sistema, haciendo que las tareas complejas requieran de un prolongado tiempo de ejecución que dependiendo de la aplicación puede ser asumible o no. Lo que induce a que los sistemas de visión artificial con respuesta en tiempo real con mínima utilización de recursos sea un campo por explorar.

La velocidad de procesamiento de la mayoría de los sistemas de visión depende directamente de la capacidad del hardware en el que están implementados. A su vez, éste depende de los avances tecnológicos que se vayan suscitando en el momento y claro está, de la capacidad del software con el que trabajan. La búsqueda en disminuir el tiempo de

procesamiento de los sistemas de visión artificial es lo que ha incitado que diferentes investigadores y universidades del mundo busquen una solución factible en cuanto a la rapidez y eficiencia. El objetivo es revertir en parte la dependencia de los sistemas de visión artificial a su hardware y a la vez hacer sistemas de mucha mayor capacidad de velocidad de respuesta, con un comportamiento cercano al tiempo real.

Por lo anteriormente mencionando, se ha llegado a la necesidad de crear sistemas de visión con un tiempo de respuesta y procesamiento de información más corto con hardware cada vez más eficiente y simple. En este contexto, los sistemas bio-inspirados y neuromórficos muestran las facultades para ser una solución óptima a la gran cantidad de aplicaciones en el entorno industrial, laboral y en un futuro, hasta en las necesidades de la vida cotidiana.

Lo que hace interesantes de investigar a los sistemas neuromórficos, y en especial a las cámaras basadas en eventos, es que proporcionan soluciones diferentes a las convencionales, reduciendo enormemente la cantidad de datos requeridos para obtener la misma información, entendiendo como dato la unidad física de almacenamiento para una información específica. Esto trae consigo ventajas adicionales como la reducción requerimientos de procesamiento, además de lograr un tiempo de respuesta más corto del que en la actualidad la mayoría de sistemas de visión convencionales basados en marcos o frames pueden alcanzar. Al requerir de menos datos para describir una misma información, necesita un hardware menos extenso en capacidades de almacenamiento de datos y con menor consumo de energía, dejando en segundo plano la dependencia de los sistemas de visión artificial a su hardware.

Es por lo anterior que en esta tesis se describen desarrollos de implementación hardware y software que pueden abrir las puertas a la incorporación de sistemas neuromórficos en la industria de la manu-

factura y de procesos, describiendo desde la generación del estímulo captado por el sistema bio-inspirado hasta la acción del dispositivo industrial. De este modo la aplicación de los sistemas bio-inspirados en la automatización industrial ofrecen nuevas perspectivas y un abanico de aplicaciones más amplio del que convencionalmente se viene haciendo.

1.3 Objetivos

El objetivo general de esta tesis es el de proponer nuevas técnicas de procesamiento de señales provenientes de sensores bio-inspirados, principalmente de cámaras de visión, y plantear escenarios que permitan la inclusión de estos sensores en el sector de la automatización industrial. Para conseguirlo, fue necesario plantear investigaciones desde la adquisición, procesamiento y transmisión de la información proveniente del sensor, hasta su incorporación en un sistema industrial y su actuación sobre elementos a controlar.

Inicialmente se hizo un estudio y recopilación de información para determinar características de las cámaras de visión *frameless* basadas en eventos, el formato de la información recibida desde los sensores asíncronos y los tipos de cámaras bio-inspiradas que existen actualmente. Se determinaron las necesidades básicas existentes de hardware en los que el método de transmisión de información adquirida en forma de eventos pueda ser eficazmente implementado.

La mayoría de los sistemas basados en eventos tienen un enfoque en la investigación, con especial énfasis en la robótica móvil. Las ventajas de este tipo de implementaciones hardware en entornos industriales, específicamente en el sector de la industria manufacturera o de procesos no ha sido explotada. En el trabajo que describe esta tesis se analizaron diversas metodologías para poder incorporar sistemas neuromórficos en un entorno industrial por lo que en los siguientes capítulos se des-

criben las ideas propuestas como punto de inicio en la incorporación de estos sistemas en la industria. Paralelamente también se fijó el objetivo de poder convertir la información que produce un dispositivo que no es basado en eventos, en información como si lo fuera, por medio de hardware simple. Comprobando compatibilidad, usos, aplicaciones, ventajas y desventajas, abriendo la posibilidad de procesados bio-inspirados a dispositivos ya instalados y que no generan eventos.

Si bien, en un sistema industrial es poco frecuente que la mayoría de las tareas de procesamiento las realice el dispositivo de instrumentación que realiza las mediciones, resulta de interés que sea el propio dispositivo el que realice un primer procesado para eliminar datos irrelevantes o extraer características básicas de la información adquirida. Se abordó la tarea de poder aplicar procesamiento sencillo in situ del propio dispositivo generador de eventos para poder transmitir la información ya depurada al dispositivo de control y que este tome las decisiones y actué. Aunado al objetivo general, el sistema desarrollado debería ser capaz de ejecutar un procesamiento en tiempo real, también de fiabilidad y flexibilidad según las necesidades de la aplicación. Además de ser probado en un entorno de exigencia industrial y compararlo con alguna solución vigente en el mercado. De tal manera se puede corroborar si las ventajas de los sistemas neuromórficos logra salir más allá del laboratorio de investigación.

Para poder cumplir con los objetivos anteriores, fue necesario cubrir un amplio marco teórico en el que se enlaza desde el sentido la vista y el ojo, los sistemas neuromórficos y su procesamiento, los sistemas de comunicación industrial y las tendencias de la industria 4.0.

Capítulo 2

Marco Teórico

¿Qué relación puede guardar el primer organismo que desarrolló sensibilidad a la luz hace 600 millones de años con el robot que empaquetó tu desayuno esta mañana? En este capítulo se presenta de forma breve, pero considerando los puntos más relevantes de un panorama general, el contexto en donde se desarrolla el trabajo de esta tesis. Se hace un recorrido iniciando con el sentido de la vista, su evolución y fisiología, su forma de transmisión, las analogías electrónicas existentes tanto en captura, transmisión y procesamiento hasta la descripción de sistemas poco explorados en cuanto a la incorporación de dispositivos bio-inspirados en el sector industrial.

2.1 La Vista y su Evolución

Los organismos que poseen el sentido de la vista, ya sea vertebrados o invertebrados, insectos o mamíferos, aún son muy superiores a cualquier dispositivo electrónico de visión, pues su relación de prestaciones contra consumo de energía en cuanto a captura y procesamiento aún están lejos de ser alcanzadas. Sin embargo, el interés por conocer cómo ha evolucionado el sentido de la vista de los seres vivos a través de la historia y la necesidad y ambición de desarrollar dispositivos análogos a los naturales hacen de este campo fuente permanente de investigaciones e innovaciones.

Charles Darwin, considerado por la comunidad científica como el investigador más influyente de la evolución biológica, realizó consideraciones sobre cómo el ojo de los vertebrados podría haber evolucionado hasta como lo conocemos hoy, considerando la hipótesis de que la naturaleza había probado distintas formas de evolución de este órgano y solo algunas habían prosperado, dando como resultado las dos principales ramas de tipos de visión biológica hasta nuestros días, las de los vertebrados, como los humanos, y las de los no vertebrados como los insectos, con similitudes pero también con muchas diferencias.

Las investigaciones científicas a lo largo de la historia en el tema de la aparición y evolución del sentido de la vista en organismos ha tenido principalmente tres ramas: la examinación de fósiles encontrados, de la estructura de los ojos en animales extintos y la evaluación de los ojos mientras se desarrolla el embrión. Además, en la actualidad hay una cuarta rama de la investigación en la que se realiza alguna comparación de las anteriores mediante comparativa molecular genética.

Según investigaciones recientes se puede aproximar que hace 600 millones de años [9] organismos evolucionaron y desarrollaron fotoreceptores capaces de producir estímulos a través de la luz y que por medio de fototaxis eludían o se acercaban a la fuente de luz, principalmente

para evadir depredadores o en búsqueda de alimento guiándose principalmente por las sombras.

To suppose that the eye, with all its inimitable contrivances for adjusting the focus to different distances, for admitting different amounts of light, and for the correction of spherical and chromatic aberration, could have been formed by natural selection, seems, I freely confess, absurd in the highest possible degree. Yet reason tells me, that if numerous gradations from a perfect and complex eye to one very imperfect and simple, each grade being useful to its possessor, can be shown to exist; if further, the eye does vary ever so slightly, and the variations be inherited, which is certainly the case; and if any variation or modification in the organ be ever useful to an animal under changing conditions of life, then the difficulty of believing that a perfect and complex eye could be formed by natural selection, though insuperable by our imagination, can hardly be considered real.

Charles R. Darwin.

Se calcula que hasta el período cámbrico, aproximadamente hace 540 millones de años, se produjo una sensible evolución en estos fotoreceptores para dar lugar a un sistema de visión, con una producción real de imágenes del entorno. Los organismos que evolucionaron de esta manera contaron como puede suponerse con ventajas y atributos sustanciales en comparación con otras especies, con mayor probabilidad de sobrevivir y tener descendencia [9].

Aunque queda completamente fuera del campo del estudio de este trabajo y la literatura es extensa, se presenta la hipótesis de cómo evolucionó el ojo desde un órgano simple hasta llegar a ser el que tienen los humanos en la figura 2.1. En esta figura se muestra en a) cómo un pequeño plato con terminaciones nerviosas es el punto de inicio del desarrollo del ojo de un vertebrado. En b) el plato anterior mostró una tendencia a lo largo del tiempo de encogerse y doblarse. En c) se muestra cómo se crearon diferentes ranuras cóncavas y con-

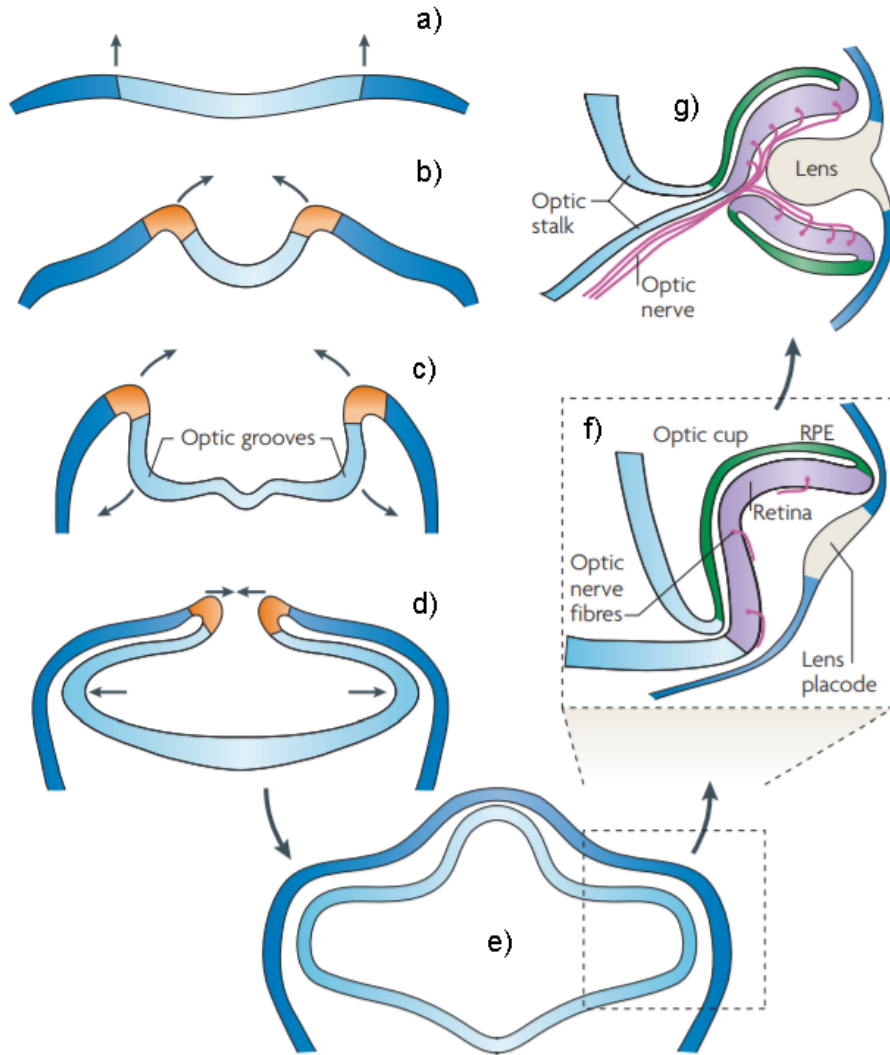


Figura 2.1: Evolución de la concavidad del ojo de un vertebrado [9].

vexas dentro de la cavidad. En d) los bordes superiores se empiezan a acercar uno al otro y se expande la cavidad creada. En e) después de que los bordes han sellado la cavidad, el cerebro del organismo crece y la vesículas ópticas crecen también hacia afuera, contactando con la superficie e induciendo la creación de una pequeña platina en forma de lente. En f) la vesícula óptica se contrae en los laterales en donde la retina aparece, y el espacio ventricular que había desaparece. Aparecen células ganglionares de la retina que envían axones fuera de la superficie de la retina. La superficie ectodérmica en la pequeña platina toma mayor forma de lente. En g) el espacio que alberga al ojo crece circunferencialmente, y encapsulando los axones del nervio óptico. Un proceso complejo, que llevó millones de años y millones de pruebas de la naturaleza hasta tomar la forma de los ojos de los vertebrados, que aunque existen también variedades, en esencia son iguales solo con particularidades más o menos pronunciadas según el recorrido que la especie de animal ha realizado a lo largo de la historia.

2.2 Funcionamiento del Ojo

El ojo de los vertebrados, sin importar la especie, ha evolucionado de la misma rama, sin embargo cuentan con pequeñas diferencias que han ido adaptando como respuesta a su instinto de supervivencia. Esta sección describe brevemente el funcionamiento del mismo.

Como se ha mencionado anteriormente, el ojo tiene unas capacidades asombrosas de las cuales la tecnología aún esta lejos de igualar. El ojo es capaz de enfocar objetos lejanos y cercanos con gran rapidez mediante mecanismos sencillos y de bajo consumo de energía. Puede obtener información bajo condiciones pobres de iluminación y puede prever una sobresaturación de los fotoreceptores ante condiciones de mucha iluminación, obteniendo siempre una resolución de imagen alta.

En primera instancia la luz pasa a través de la córnea hasta proyectarse en el fondo de la cavidad donde está alojada la retina, esta proyección llega de manera inversa por el efecto de pasar a través del cristalino, tal como lo hace una cámara convencional, el efecto se muestra en la figura 2.2.

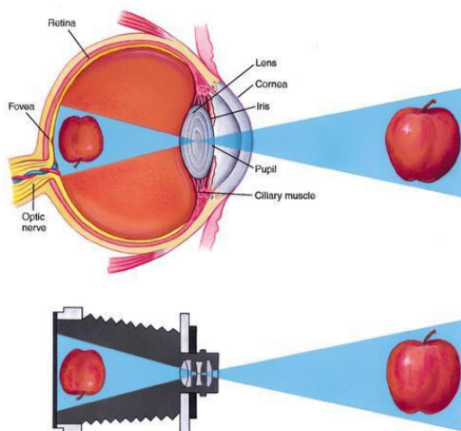


Figura 2.2: Simil del funcionamiento del ojo a una cámara [10].

El punto de proyección del objeto que se observa en la retina es debido a que en esta zona están alojadas células fotorreceptoras, que transformarán las diferentes longitudes de onda en pulsos eléctricos dirigidos al nervio óptico para que posteriormente los procese el cerebro. Estas células tienen el nombre de conos y bastones. Las primeras son las encargadas de responder al color y las segundas a estímulos según la intensidad de iluminación. Este par de células son una de las principales diferencias entre el sentido de la vista entre animales vertebrados, en el que influyen la cantidad que la especie tiene de cada uno y la predisposición, para poder ver mejor a larga distancia, en condiciones de poca iluminación o el rango de colores a distinguir.

Esta proyección se enfoca en la retina en una pequeña porción de la misma llamada mácula, representada en la figura 2.3, como una depre-

sión en la superficie de la retina que en su parte con mayor depresión es conocida como fovea.

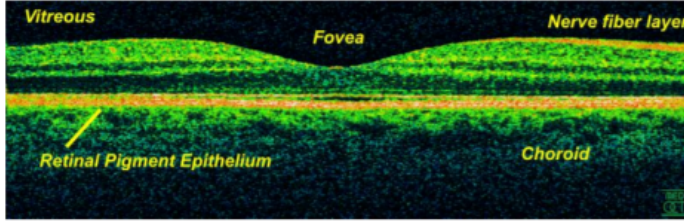


Figura 2.3: Partes de la mácula [10].

Un punto importante de las cualidades de las diferentes partes y funcionalidades del ojo se plantea en [10], en donde se establece que para poder obtener la amplia resolución angular del ojo, considerando el espacio tan pequeño de proyección y las zonas donde se proyecta la imagen en la retina que carecen de células fotorreceptoras, el ojo ejecuta pequeñas oscilaciones para rellenar espacios entre células y completar la imagen. Sin embargo, parece muy complicado que el cerebro canalice correctamente la señal recibida que se está proyectando continuamente en la retina si el ojo está moviéndose constantemente. Este factor, sumado al producido por las reacciones químicas y físicas de la fotorrecepción como la difracción del rayo de luz al atravesar la córnea puede causar una imagen poco enfocada. Por lo que se cree que para poder obtener la imagen adecuadamente, se debería tener al menos una neurona por cada fotorreceptor, y cada neurona conectada a varias células fotorreceptoras. Con esto se consigue recomponer la imagen desenfocada y la ventaja de tenerla de manera redundante. Es por esto que en la siguiente sección se describe a grandes rasgos la neurona y su funcionamiento.

2.3 La Neurona

La neurona es una de las células más complejas del cuerpo humano, pues su sistema interno y su interconexión son capaces de procesar información a través de reacciones químicas y diferenciales de potencial. Se estima que el cuerpo humano posee alrededor de 86 mil millones de neuronas con 1.5×10^{14} de conexiones y que en el cerebro hay alrededor de 20 mil millones de neuronas [11]. La neurona tal como se representa en la figura 2.4, tiene tres elementos principales: soma, axón y dendrita.

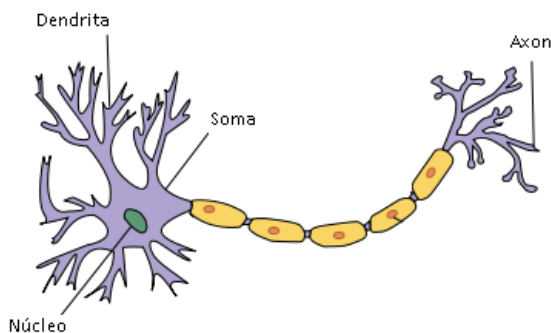


Figura 2.4: La estructura más común de la neurona [12].

La soma es la parte central de la neurona y contiene al núcleo entre otros organelos en donde procesa principalmente la señal electroquímica que recibe y que transmitirá el estímulo a otras células, además de cumplir con todas las funciones básicas del núcleo de una célula, necesarias para su existencia. Del soma se extienden las dendritas, las cuales están conectadas a neuronas adyacentes vía enlaces llamados sinapsis. Las dendritas son la entrada de estímulos al núcleo para su procesamiento. Los axones son la contraparte de la dendrita, pues son la salida del procesamiento del soma, y aunque la neurona puede tener solo un axón, este puede estar conectado a más de una dendrita de otra célula.

El estímulo se transmite de neurona a neurona a través del incremento de potencial almacenado en la célula, estimulando diversos canales químicos como el ion Na^+ en la membrana. El potencial se recorre a través de todo el axón induciendo que se abra un canal de iones K^+ para pasar la energía de una neurona a otra u otras, reiniciando el potencial en la neurona transmisora. El axón está cubierto usualmente por mielina, una sustancia que ayuda a incrementar la velocidad de propagación del estímulo.

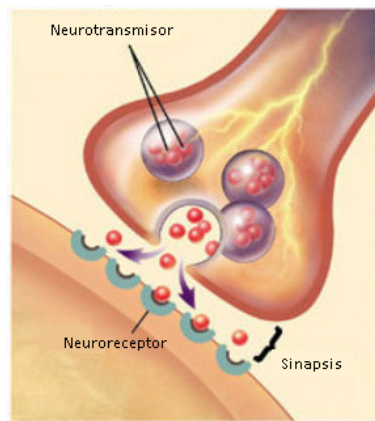


Figura 2.5: Representación de la transmisión de un impulso de una neurona a otra[12].

En la figura 2.5 se muestra la sinapsis y cómo son liberadas las sustancias químicas para poder crear el puente virtual que transmitirá el estímulo. Las terminales del axón contienen moléculas de neurotransmisores, que con la carga de potencial de la neurona se liberan hacia el espacio entre el axón y la dendrita de la siguiente neurona, comúnmente llamada célula postsináptica, en la cual se ubican los neuroreceptores que capturan las moléculas liberadas dando paso así a la transmisión del estímulo de una célula a otra.

El comportamiento descrito, como se puede intuir, es un proceso complejo de describir en una formulación, pero un modelo que es usual-

mente aceptado, y que constituye una base de muchos trabajos es el modelo de Hodgkin y Huxley, biólogos que realizaron experimentos con la neurona del calamar gigante. En base a sus conclusiones obtuvieron un modelo eléctrico de la neurona que puede ser expresado matemáticamente [13].

En su trabajo consideran la membrana de la célula como un circuito donde hay una corriente que está cargando un capacitor en paralelo a una resistencia que representan los canales de la membrana. Tal como se muestra en la figura 2.6.

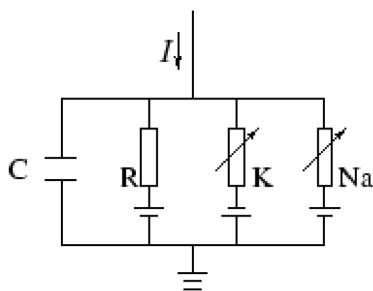


Figura 2.6: Simplificación de la circuitería equivalente de la membrana de una neurona [13].

La corriente fluye a través de la membrana del canal de sodio, del canal de Potasio y un canal de fuga general. Por lo que el comportamiento de su potencial puede ser representada con la ecuación 2.1 .

Aunque la ecuación 2.1 no tiene algún límite o frontera y podría generar una respuesta similar a la de la neurona a cualquier frecuencia, la neurona real tiene límites de operación que están en el rango de -80 a 40mV, 1 a 20nA y una frecuencia de disparo que va de 1 a 200 Hz.

Una vez que se obtuvo un modelo matemático experimental que representa el comportamiento de uno de los mecanismos más complejos de

la naturaleza, se puede entender el interés de la comunidad científica por desarrollar y crear sistemas bio-inspirados.

$$C \frac{du}{dt} = - \sum_{k=1}^3 I_k(t) + I_C(t) \quad (2.1)$$

2.4 Sistemas Neuromórficos

Carver Mead reconocido investigador del Instituto Tecnológico de California, introdujo el término *Neuromórfico* en un artículo de principios de la década de 1990 [14].

I do not believe that there is any magic in the nervous system—that there is a mysterious fluid in there that is not defined, some phenomenon that is orders of magnitude more effective than anything we can ever imagine. There is nothing that is done in the nervous system that we cannot emulate with electronics if we understand the principles of neural information processing.

Carver Andress Mead

En este artículo Carver se pregunta qué tipo de primitivas computacionales limitadas por la física de los dispositivos era análoga de alguna manera al sistema nervioso. Distinguió que en ambos casos el estado de las variables es analógico, es decir, continuo, representado por una carga eléctrica. Para construir un sistema nervioso en un sistema computacional se deben poder crear conexiones específicas en donde una determinada salida está conectada a entradas específicas, a la vez de que necesita poder estar aislada eléctricamente de nodo a nodo, de origen a destino. Tanto en los sistemas eléctricos como en el nervioso, el aislamiento se consigue construyendo una barrera que no permita que se filtre energía a otro nodo cercano al que se está cargando. En

el sistema nervioso la barrera es construida por la diferencia dieléctrica entre la grasa y las soluciones acuosas que rodean a las células, mientras que en los sistemas eléctricos se logra con el espacio entre el silicio y el dióxido de silicio.

Por lo anteriormente descrito, Carver considera como sistema neuromórfico a cualquier dispositivo que es posible ser construido de tal manera que puede implementar las mismas operaciones básicas que tiene el sistema nervioso, además de la posibilidad de construir sistemas enteros basados en los principios de organización del sistema nervioso.

Tobi Delbruck, que realizó investigaciones junto con Carver, en su trabajo [15] define la posibilidad de representar una retina a base de elementos de silicio. La arquitectura biológica en la que se basaron es la que se muestra en la figura 2.7

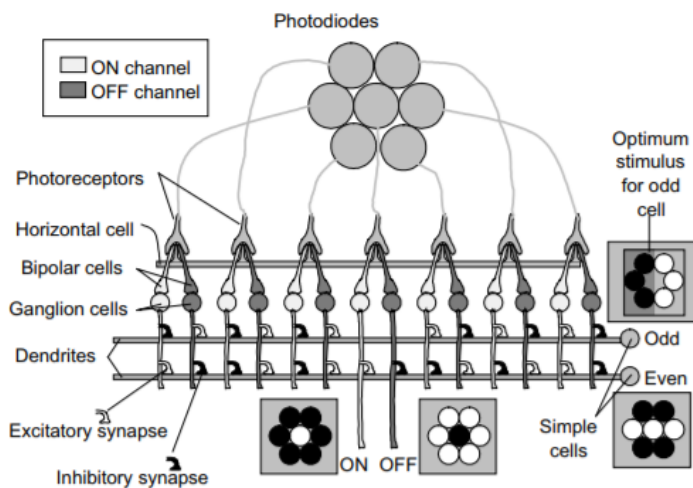


Figura 2.7: Esquema análogo de un ojo implementado en el chip de una cámara basada en eventos [15].

Para la implementación de esta arquitectura se hicieron algunas asunciones para simplificar los circuitos de la retina pero que eran suficientes para producir un resultado realista a un estímulo visual. Dentro de la retina de silicio la escena visual está enfocada en un arreglo circular de fotodiodos que representan la etapa de transducción de los fotorreceptores del ojo. En la figura 2.8 se muestra un esquemático simplificado de la analogía con la retina y la neurona. En a) la cadena de circuitos analógicos de cada píxel que implementa la retina, en b) se muestra el circuito de la sinapsis y en c) el axón.

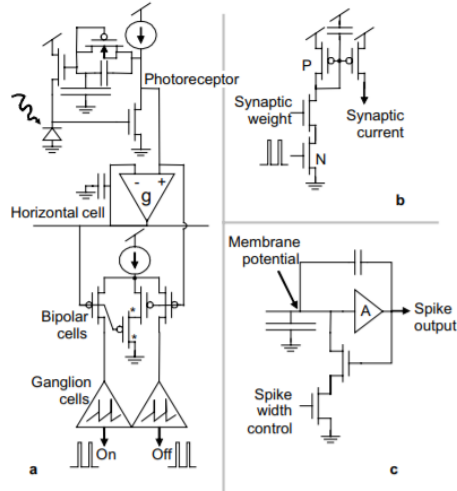


Figura 2.8: Esquemático simplificado de una unidad de la retina propuesta en [15].

En la figura 2.9 se muestra la respuesta obtenida en [15] para el modelo de la retina de silicio. En a) está representada la respuesta a un incremento en la iluminación global, y en b) se muestra la respuesta de la retina ante un punto destellante en el píxel central.

Considerando que la respuesta mostrada en la figura 2.9 es solo la de una unidad, fue necesario crear una técnica para poder encapsular esta información y transmitirla a otros dispositivos. La metodología de codi-

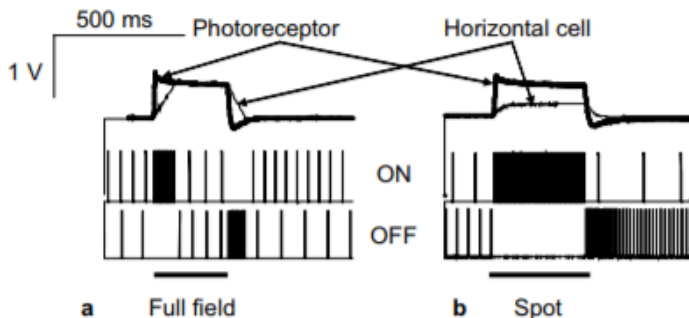


Figura 2.9: Respuesta obtenida de una unidad de la retina propuesta en [15].

ficación utilizada desde principios de los años 1990, fue la denominada *AER (Address Event Representation)* o en español, *Representación de Eventos por Direccinamiento*, con el objetivo inicial de intercomunicar diferentes sistemas electrónicos neuromórficos [16].

Uno de los campos mejor desarrollados con esta técnica de codificación es el campo de la visión artificial, en donde las cámaras producen datos en forma de eventos, asíncronos sin necesidad de ningún disparador, ejemplos de esta tecnología son [17], [18] y [19] principalmente. Los datos son generados cuando hay una diferencia en la intensidad de luz recibida por cualquiera de los sensores o píxeles que están dispuestos en una matriz. Cada píxel de la cámara será sensible a la diferencia de la intensidad de luz y este producirá un evento si esta diferencia es mayor a un límite predefinido. Los eventos generados incluyen información acerca de la dirección de cada píxel en el sensor, donde el límite, normalmente previamente configurado por hardware o software, fue excedido junto con el tiempo en el que ocurrió, para así generar un único evento tanto en espacio como en tiempo. Además, es posible definir si el evento es causado por el incremento o decremento de intensidad de luz recibido en dicho píxel, causando un evento positivo o negativo. Este comportamiento es similar a cómo procesa la informa-

ción el cerebro de los mamíferos [20] lo que hace interesante utilizar esta metodología para procesamiento de información, filtración y extracción de características, detección de movimientos en escenas [21], [22].

Una de las principales características que hacen diferencia entre una cámara basada en eventos de una convencional, es el sistema de iluminación para la captura de la escena. La iluminación apropiada según las características que se requieran extraer es indispensable, lo que hace que este complemento de las cámaras sean un factor clave en los sistemas de visión industrial tradicionales basados en fotogramas, sin embargo, mantener una luminosidad constante normalmente es difícil debido a que el ambiente puede ser propenso a cambios no controlables. Las soluciones tradicionales para este problema requieren tipos específicos de luz según la aplicación, ejemplos de trabajos realizados en la importancia de la iluminación son [23], [24] y [25]. Por otro lado, las cámaras basadas en eventos minimizan los efectos de los cambios de iluminación a lo largo del tiempo debido a que el píxel considera solamente los cambios de intensidad de la luz independientemente de las condiciones del ambiente.

La siguiente sección hace una breve presentación la metodología de codificación de la información producida por sistemas neuromórficos.

2.5 Representación de Eventos por Direccionamiento, AER

Las neuronas comunican largas distancias, con respecto a su tamaño, utilizando una representación de la información basada en pulsos. Se podría discernir que debe haber un gran gasto de energía para transportar esta información, sin embargo, las observaciones e investigaciones hechas en este campo demuestran lo contrario. Los sistemas neuronales utilizan el principio de la conservación de la energía, redu-

ciendo el total del número de pulsos para transmitir la información de la manera más eficiente posible. Esto provoca que la unidad receptora sea la encargada de integrar todos los pulsos que va recibiendo a lo largo del tiempo. Adaptación temporal, inhibición lateral y correlación de los pulsos son ejemplos de procesamientos neuronales que resultan interesantes para el ahorro de energía [26].

En el pulso que genera la neurona, la altura del pulso no representa información adicional, sino el tiempo en el que se ha generado; la neurona transmite información solo al momento en que el impulso comienza. Este proceso se realiza de manera asíncrona, lo que significa que el evento se puede dar en cualquier momento sin importar ningún reloj o disparo de captura. Una manera de representar esto sería crear un listado de eventos precisando el momento exacto en el que el pulso ha sido recibido, y el origen de donde ha sido producido. Esta idea sumada al trabajo de J. Lazzaro [27] fijaron los primeros cimientos del protocolo de codificación de eventos generados por sistemas neuromórficos. La idea del listado de tiempo se expandió a la posición y conforme han evolucionado las prestaciones de los sistemas neuromórficos se han ido incorporando distintas capacidades de transmisión de información, por lo que al igual que los sistemas biológicos y neuromórficos, el método AER también ha evolucionado.

En 1993 T. Delbruck [28] sugirió un primer estándar del protocolo de codificación AER, llamado AER 0.2, por ser una versión de prueba y a la espera de la aceptación de los demás grupos de investigación que por entonces trabajaban con sistemas neuromórficos. Este protocolo tenía un mayor enfoque de la parte hardware que de la propia codificación de la información. Era una conexión punto a punto, unidireccional asíncrona entre un emisor y un receptor. La conexión entre emisor y receptor consistía en dos tipos de cables: control y datos. Los datos eran controlados exclusivamente por el emisor y sentidos por el receptor. La codificación de la información dependía completamente de la aplicación que se implementaba. Esta última condición se

mantiene hasta ahora pues, aunque el protocolo ha evolucionado en diferentes versiones, cada vez más la parte software restringe o facilita la implementación de una u otra versión según la aplicación final y las cualidades requeridas.

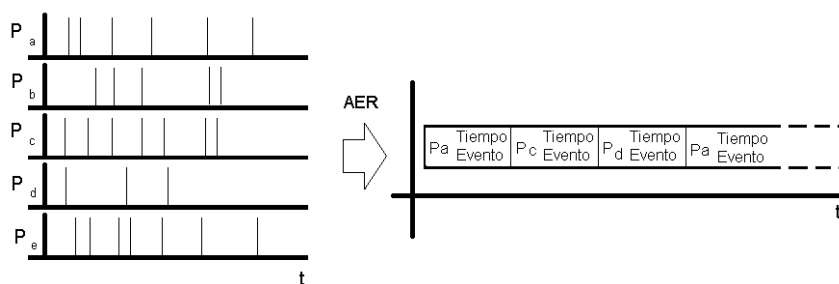


Figura 2.10: Esquema básico de codificación de eventos mediante AER. Del lado izquierdo se representan eventos en diferentes posiciones a, b, c, d y e, del lado derecho la manera en la que se codificarían los eventos en todas las posiciones propuestas, según la generación cronológica del evento.

En la figura 2.10 es representado un esquema básico de la manera en la que el protocolo de codificación AER actúa. En un sistema de visión basado en eventos, las diferentes fuentes de eventos serían cada uno de los píxeles, los cuales serían codificados en posición y en el tiempo de aparición.

AER es una metodología de codificación interesante y probada para el intercambio de información entre sistemas neuromórficos. En este trabajo, uno de los planteamientos principales es poder aprovechar esta codificación en sistemas que requieren las cualidades de los sistemas neuromórficos: reducción del consumo de energía y reducción de la cantidad de datos a transmitir para una misma información. El efecto de estas condiciones origina una reducción del tamaño de hardware de procesamiento y memoria empleada.

2.6 Procesamiento de información mediante eventos

Típicamente la información generada por sistemas basados en eventos es tratada por Redes Neuronales Pulsantes o mejor conocidas como SNN por sus siglas en inglés. Las SNN fueron propuestas por primera vez en [29], en donde clasifica este tipo de redes neuronales como de tercera generación. Define como redes neuronales de primera generación aquellas que solo procesan señales binarias, es decir OFF/ON o 0/1 y por supuesto, solo estos valores posibles de salida. La segunda generación procesa valores continuos a su entrada para producir de igual manera valores continuos a su salida. Esta salida representa la tasa de disparo de la neurona de salida, lo que es más aproximado a la realidad de una neurona desde el punto de vista neuromórfico de lo que son las redes de primera generación. Las redes neuronales de tercera generación codifican la información en el tiempo en el que producen un disparo, impulso o evento a la salida, este evento o disparo se produce al integrar a lo largo del tiempo el valor de su entrada. Por esta razón, las SNN son las redes más comunes para el procesamiento de eventos, al considerar que la parte importante de la codificación de la información está en el espacio de tiempo entre impulsos y no en sí en magnitud del impulso. El valor de integración al que la neurona disparará, es decir, el umbral de disparo, es el parámetro a entrenar en este tipo de redes neuronales.

También existen trabajos en los que se procesa la información de eventos con Redes Neuronales Convolucionales o *CNN*. Las *CNN* son usualmente utilizadas en los sistemas convencionales de visión artificial, no obstante, también han resultado de interés en el tratamiento de eventos, especialmente en los sistemas de reconocimiento de objetos [30], [31].

Hoy en día, las técnicas de procesamiento basado en eventos que se están desarrollando en diversas universidades y centros de investigación tienen un marcado enfoque hacia el reconocimiento de patrones

utilizando sistemas neuromórficos y aprendizaje automático, [32], [33] y [34] son solo algunos, pues la literatura en lo que respecta a dispositivos basados en eventos con este enfoque es extensa.

El procesamiento de datos de sistemas bio-inspirados se encuentra en expansión por lo que se consideró que uno de los campos en crecimiento que puede beneficiarse de este tipo de sistemas es el sector de la fabricación automatizada, y más concretamente en lo que respecta a sistemas basados en eventos en cuanto al intercambio de información se refiere. Este hecho es el motivo por el cuál se ha hecho una búsqueda de las características, tendencias y necesidades del sector de las comunicaciones en la industrial de la automatización, y cómo estos sistemas basados en eventos pueden ayudar al desarrollo del sector industrial.

2.7 Comunicaciones Industriales

Las comunicaciones industriales surgieron a la par que los dispositivos de control industrial, en los años 70, pues rápidamente surgió la necesidad de compartir información de supervisión y control entre los equipos en planta. Pero fue esta necesidad y la falta de estandarización lo que originó que durante casi tres décadas los distintos fabricantes de equipos utilizaran diferentes metodologías de transmisión y formas de codificar la información, que hacía muy complicado poder enlazar equipos de distintos fabricantes. Los fabricantes, ya sea que utilizaran circuitos integrados en común o crearán los propios, empeñaban sus esfuerzos en hacer diferentes protocolos como mera estrategia de mercado. Las tecnologías más comunes para la transmisión física de datos eran RS-232, RS-422, RS-485 y CAN principalmente. Dentro de cada tecnología era común encontrar las distintas formas de codificar y actualizar la información, es decir los protocolos de comunicación. Entre los más importantes se encuentran Modbus RTU, FINS, Hostlink, Toolbus, Devicenet, Profibus, Interbus, HART, ControlNet y AS-i. Es entendible que surgiera la necesidad de una estandarización para sol-

ventar esta torre de Babel en la industria. El primer paso fue con la aparición del modelo OSI a mediados de los 80. Este modelo trataba de marcar la referencia de cómo debería enviar, codificar, autentificar, decodificar y utilizar la información por medio de siete capas, figura 2.11. Pero fue hasta finales de la década de los 90 cuando las demandas por parte de los usuarios a la compatibilidad de equipos en planta orilló a los fabricantes a utilizar una misma capa física como medio de transporte: Ethernet.

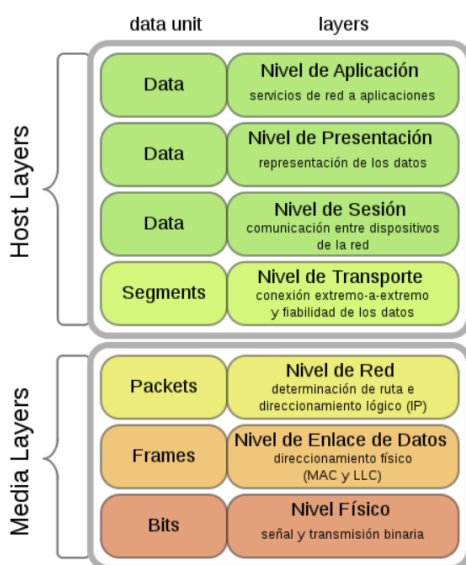


Figura 2.11: Capas del Modelo OSI. Divididas en dos grupos: las capas dependientes del medio de transporte y las capas del dispositivo que envía o que recibe la información [35].

2.7.1 Comunicaciones Industriales en la Actualidad

La cantidad de datos que se transmiten a través de una red de comunicaciones se incrementa conforme se dotan de mayores capacidades e inteligencia a los dispositivos periféricos de la red. Cada vez es más interesante poder disponer de mayor cantidad de información para así

tomar las decisiones pertinentes. Especialmente en ambientes industriales donde la respuesta en tiempo real y baja latencia son requeridas. La saturación de las redes de comunicación debido a que se han incorporado equipos avanzados que generan y transmiten una gran cantidad de datos puede ser un problema [36]. Este incremento se ha visto propiciado por las innovaciones tecnológicas en comunicaciones en las últimas tres décadas, pero sobre todo en los últimos años con el impulso de inversión en este sentido que ha traído el paradigma de la industria 4.0, industria interconectada o el Internet Industrial de las Cosas, todos estos términos hacen referencia a una misma idea y se utilizan según la posición geográfica en la que se encuentre: Europa, Estados Unidos, Asia o América latina. Este concepto tiene como principal objetivo la optimización de los recursos disponibles para crear mayor beneficio utilizando la mayor cantidad de datos posibles en todo el ciclo de vida de un producto, considerando desde las preferencias del consumidor en cierta zona geográfica dadas ciertas condiciones climatológicas, hasta el mantenimiento preventivo del rodamiento de un ventilador, tratando de cerrar el círculo productivo de cualquier producto o servicio. Como se puede intuir, la cantidad de datos a procesar y transmitir puede ser enorme según los alcances de la aplicación. En consecuencia, se está evolucionando constantemente hacia redes con mayor capacidad de transmisión, pero también se ha puesto interés en que los dispositivos que transmitan y procesen información codifiquen en un menor tamaño de datos la mayor cantidad de información.

Por otro lado, las principales aplicaciones implementadas con cámaras basadas en eventos han sido desarrolladas con propósitos de investigación, emulando un sistema neuromórfico. Algunos trabajos se centran en el desarrollo y mejora de sistemas de intercambio de datos de dos o más dispositivos bio-inspirados [37] y [38], pero poco se ha indagado en otros sectores en donde la metodología de producción de datos, inherente a esta tecnología, pudiera beneficiar. Uno de estos sectores es el de la industria de la automatización. Los sistemas basados en even-

tos todavía no han logrado tener popularidad en el ámbito industrial pues están diseñados para trabajar en conjunto con otros dispositivos también de base bio-inspirada, motivo por el cual una adecuación previa a la transmisión pueda ser necesaria.

La cantidad de datos que producen usualmente los sistemas de visión basados en eventos, aunque es menor que los sistemas basados en fotogramas, puede no resultar suficiente según la aplicación a implementar, pudiendo requerir de conexiones específicas [39]. Aunque usualmente el ancho de banda que utilizan los sistemas basados en eventos es alto, pues las investigaciones se han centrado en el entrenamiento de sistemas inteligentes, en donde la cantidad de datos para representar una misma información de manera redundante es primordial. No obstante, estos datos pueden ser filtrados con el objetivo de ser más interesantes para aplicaciones industriales, obteniendo de esta manera solo una cantidad mínima útil de datos que represente la información capturada.

Los sistemas de comunicación industrial no solo necesitan un bajo ancho de banda, sino que también es necesario tener un tiempo de respuesta corto cercano al tiempo real. La carga computacional en un sistema industrial requiere un bajo consumo de energía y un eficiente almacenamiento de datos, factor determinante en sistemas descentralizados. Las características que debe poseer una red de comunicación industrial se explican en la siguiente subsección.

2.7.2 Características de las Comunicaciones Industriales

Lo que hace diferente a una red de comunicación convencional como la que encontramos en nuestros teléfonos móviles al realizar una llamada por medio de datos, al ver un vídeo en internet o en subir una fotografía algún servicio de hosting en la nube, con una red industrial radica esencialmente en tres conceptos: la *latencia*, el *jitter* y el *determinismo* del protocolo de comunicación.

La *latencia* se refiere al retraso que toma a un bit transmitirse desde el dispositivo fuente hasta el origen, lo que se traduce en cuánto tiempo la red mantiene un dato en la red circulando por todos los dispositivos intermediarios hasta llegar a su destino final. Los factores que principalmente intervienen son: la distancia, los errores de transmisión, la recuperación de los mismos, la congestión y la capacidad de procesamiento de los dispositivos intermediarios. El *jitter* es la variación de este retraso usualmente ocasionado por la variabilidad de la disponibilidad del acceso a la red. Y finalmente el *determinismo* es la característica que tiene el protocolo para asegurar que el dato puede llegar a su destino dentro un tiempo máximo conocido.

Aunque estas características son las más importantes, cada diferente protocolo industrial las acatará de una u otra manera para lograr diferentes desempeños según la aplicación. En aplicaciones donde las prestaciones requeridas son tolerantes, como puede ser sistemas de supervisión, el determinismo es importante pero suele ser en rango de tiempos fácilmente alcanzables; por el contrario, en aplicaciones con requerimientos más exigentes, como lo son en el control de ejes o la visión artificial, el determinismo es esencial pues se trata siempre de minimizar el tiempo de ejecución.

A continuación, se da una visión general de las tecnologías de ethernet industrial más importantes. Se muestran los principios técnicos de los diversos enfoques y se intenta ponerlos en perspectiva de sus ventajas y desventajas entre los más implementados a nivel industrial [40].

2.7.3 Redes Industriales Ethernet

Una clasificación interesante, que se puede encontrar en la bibliografía que compara los distintos protocolos de comunicación que utilizan ethernet, es la que engloba los protocolos en tres clases diferentes A, B y C según el estándar IEC 61784-2 [41], [42].

La clase A utiliza hardware Ethernet estándar no modificado, así como librerías de software TCP/IP estándar para la comunicación de procesos. El hecho de que sea hardware Ethernet no modificado simplifica la integración de dispositivos, ya sea de nodos finales o de intermedarios. En algunas ocasiones se añaden pequeñas adaptaciones que proporcionan un mejor rendimiento. El rendimiento en tiempo real está limitado por retrasos impredecibles en los componentes de la red, así como de los enrutadores, switches y demás dispositivos de red.

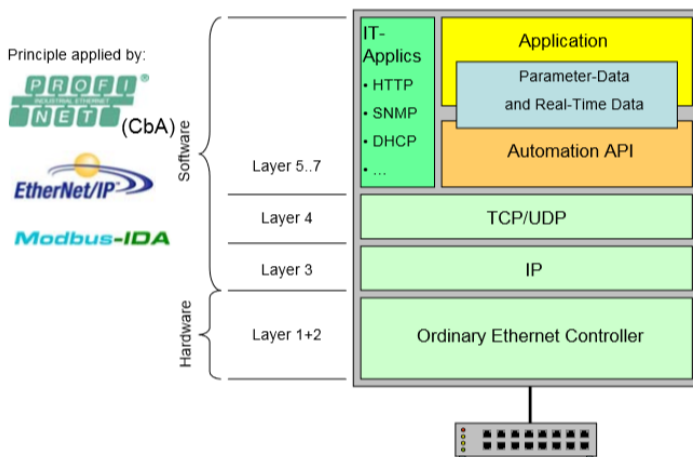


Figura 2.12: Aproximación clase A. Se utiliza hardware ethernet estándar, con la ventaja de la compatibilidad entre distintas redes ethernet, pero con aplicaciones en tiempo real limitadas [42].

La clase B utiliza hardware estándar no modificado, pero no usan TCP/IP para la comunicación de datos de proceso. Se introduce un protocolo de datos de proceso dedicado, que puede incorporar las librerías de software de TCP/IP, normalmente por temas de compatibilidad con otras redes. Normalmente, el acceso a la red Ethernet está controlado y limitado por zonas de tiempo, en el que se le da paso a la red a cada nodo.

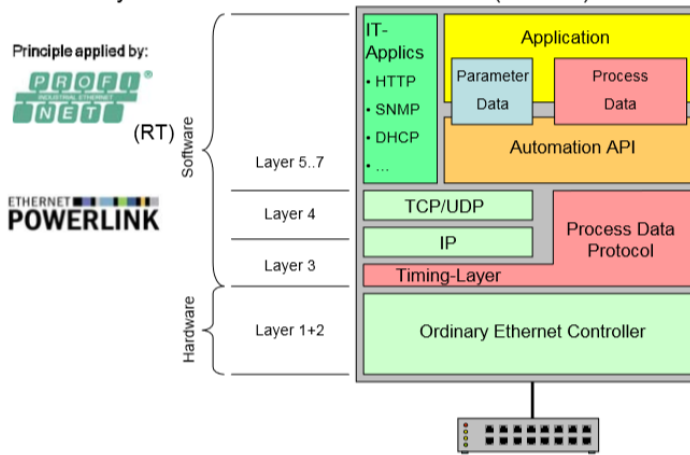


Figura 2.13: Aproximación clase B. Al igual que la clase A utiliza hardware no modificado estándar de ethernet, pero el acceso a la red está controlado y limitado, para evitar retardos y reducir los tiempos de respuesta y actualización [42].

Finalmente, la clase C tiene un claro objetivo, el rendimiento. Se debe usar hardware dedicado (al menos en el lado del dispositivo esclavo o receptor). Al igual que en la clase B no se excluye el uso de TCP/IP pero no es la base del protocolo. La compatibilidad con otros protocolos en una misma red usualmente es complicada.

2.7.4 Principales Protocolos de Comunicación basados en Ethernet

A continuación se hará una breve introducción con las principales características de los protocolos de comunicación industrial basados en ethernet [43] y [44]. Los protocolos que abarcan la mayor parte del mercado son Ethernet/IP, Profinet, Modbus TCP, Ethercat y Powerlink.

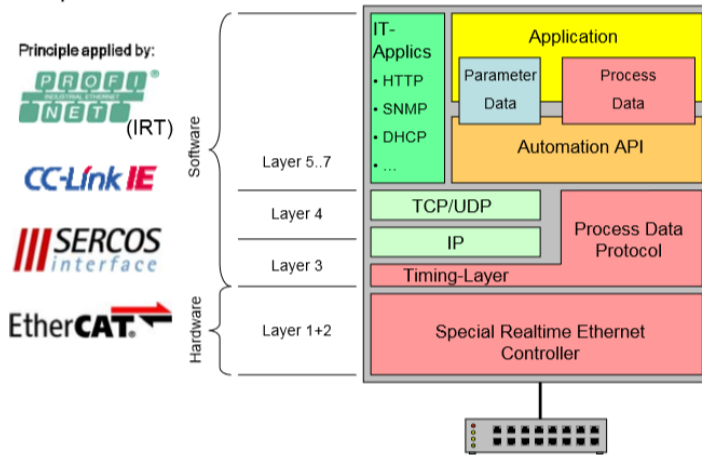


Figura 2.14: Aproximación clase C. El rendimiento es su objetivo, por lo que no solo utiliza hardware dedicado, sino que requiere capas de software no estándar TCP/IP [42].

Ethernet/IP

Las letras IP vienen de Protocolo Industrial, comúnmente confundidas con las siglas de TCP/IP que aquí se refieren a Protocolo de Internet. Este protocolo está respaldado por la ODVA (Open DeviceNet Vendor Association) por lo que comparte las capas de aplicación con Devicenet y ControlNet. Tiene implementado el protocolo CIP [45] en las capas superiores tal como se muestra en la figura 2.15.

Se basa en el modelo productor-consumidor por lo que es muy eficiente al momento de transmitir datos entre los diferentes nodos de una red. No obstante, esta característica requiere que los dispositivos enrutadores estén correctamente configurados para evitar el bloqueo de los puertos. Ethernet/IP distingue conexiones CIP de TCP/UDP. Una conexión CIP transfiere datos entre dos o más dispositivos que tengan alguna aplicación CIP en ejecución y cada una de estas conexiones puede ocupar una conexión TCP o UDP. Dependerá de las capacidades de cada dispositivo la cantidad de conexiones a poder

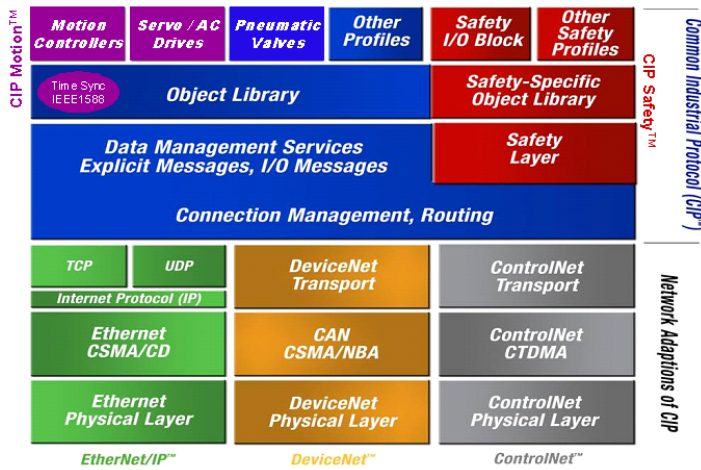


Figura 2.15: Implementación de Ethernet/IP [46].

implementar para un mismo dispositivo. El enfoque del protocolo es de clase A, pudiendo ser implementado en hardware estándar, con tiempos de actualización de ciclo alrededor de 10 milisegundos. No obstante, se puede reducir este tiempo si se utiliza la versión mejorada de CIP, llamada CIP Sync, en donde se utiliza hardware Ethernet modificado en el que se pueden alcanzar tiempos de ciclo menores, que rondan el milisegundo.

Profinet

Protocolo principalmente utilizado por la empresa SIEMENS AG, y por la organización Profibus. Existen tres versiones de Profinet (figura 2.16):

Versión 1: También conocida como CBA o Automatización basada en el componente posee ciclos de actualización adecuados para la comunicación a las capas superiores de la pirámide de la automatización, pero muy altos para las partes bajas, rondando los 100 milisegundos. Tiene un enfoque de clase A.

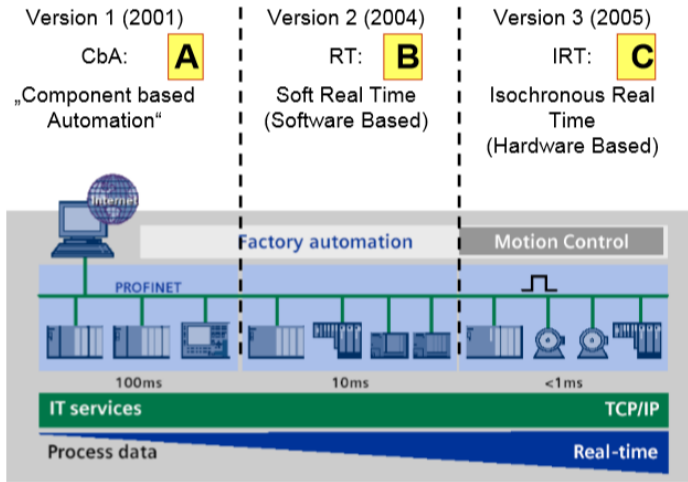


Figura 2.16: Versiones de Profinet [42].

Versión 2: Conocida como RT o Tiempo Real. Es con diferencia la versión más distribuida e implementada de Profinet. Su tiempo de ciclo ronda los 10 milisegundos. Tiene un enfoque de clase B.

Versión 3: Usualmente llamada IRT o Tiempo Real Isócrono. Es la versión con menor tiempo de ciclo, el hardware Ethernet es modificado en toda la subred en donde se instale. Su implementación en hardware especializado hace que se alcancen tiempos de ciclo menores a un milisegundo. Es posible sincronizar todos los nodos de la red, de utilidad en el control de movimiento. Su enfoque es de la clase C.

Modbus TCP

Es la implementación del protocolo serie Modbus RTU en Ethernet [47]. Es uno de los buses con más implementación a nivel mundial, sobre todo en capas medias de la pirámide de la automatización, a pesar de que no tiene una solución para transferencia de datos en tiempo real. El tiempo de ciclo depende de la aplicación en ejecución que produce los datos más que del propio hardware que lo implemen-

ta, usualmente ronda los 100 milisegundos, aunque se pueden alcanzar tiempos menores dependientes del dispositivo que lo ejecuta. La programación y configuración es de las más simples de las expuestas en esta tesis, aunque esta característica hace que los datos leídos o recibidos requieran ser interpretados de manera explícita para reconocer tipo de dato y cantidad de bytes. Es posible leer o escribir a nivel de bit o a nivel de palabra (16 bits). El enfoque es de clase A, es un protocolo muy maduro, probado y aceptado y no parece que surgirá alguna nueva versión. Difícilmente desaparecerá pues es usualmente la versión más económica de un protocolo de comunicación basado en Ethernet.

Ethercat

Este protocolo tiene un enfoque clase C en el esclavo y un enfoque clase A en el maestro. Implementa lo que es conocido como *Processing on the fly* o Procesamiento al Vuelo (figura 2.17), que quiere decir que el esclavo ya está procesando el paquete de información antes de que este termine de llegar. Requiere de hardware modificado basado en FPGA además de que cada dispositivo puede ir sumando información al paquete o datagrama que va recibiendo y enviándolo al siguiente nodo. Se puede alcanzar ciclos de actualización de datos de hasta 125 microsegundos, aunque la topología y la cantidad de nodos en la red podrían perjudicar este valor. Es una de las redes más flexibles permitiendo conectar hasta 65535 nodos y prácticamente cualquier cantidad de ramificaciones en la red.

Powerlink

Introducido por la empresa B&R, cuyo mercado principal es la producción de equipos de control de movimiento [48], se puede intuir cuáles son sus prestaciones. El control de acceso a la red se hace por POLLING, que significa que un dispositivo maestro debe dar el paso de

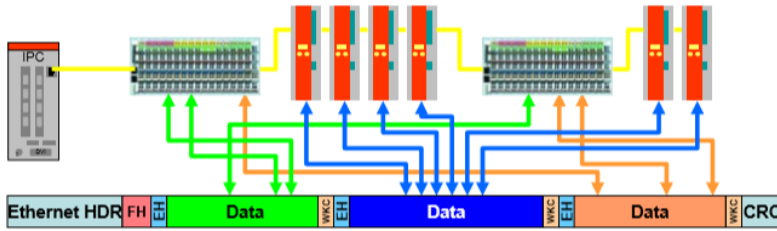


Figura 2.17: Ejemplo del procesamiento al vuelo, en donde el paquete que envió el maestro es modificado conforme va pasando a través de los diferentes esclavos de la red [42].

acceso a la red mediante el método de división de tiempos conocido como *slicing*. Aunque en sus principales áreas de aplicación no lo suelen requerir, para implementación de topologías diferentes al común *daisy chain* es necesario utilizar Hubs.

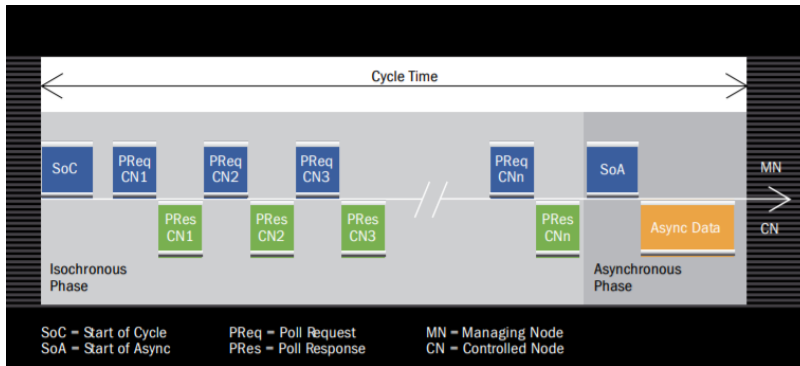


Figura 2.18: En esta figura se muestran las diferentes etapas en relación con el tiempo, en donde el nodo maestro o MN designa los momentos en los que cada dispositivo puede acceder a la red [48].

El desempeño de la red depende de la implementación del hardware en los esclavos y la topología, y también dependerá de esta última también la máxima velocidad de actualización de datos. El modelo de la transmisión de datos es maestro/esclavo con modificaciones para

que los esclavos puedan compartir información entre sí, así como con dispositivos fuera de la red. Esto se hace a través de la funcionalidad de este protocolo llamada TIME SLICING o Particiones de Tiempo (figura 2.18), en la que se tienen predefinidos los tiempos en los que el maestro gestiona el intercambio de los datos con los esclavos, y deja cíclicamente un periodo de tiempo para que los servicios antes mencionados tomen paso a la red, llamada zona asíncrona. Dependiendo de los dispositivos de la red podrían alcanzarse hasta 400 microsegundos de tiempo de ciclo.

Capítulo 3

Codificación en Eventos para Sensores no Basados en Eventos

En la actualidad, la mayoría de los dispositivos basados en eventos tienen una marcada tendencia neuromórfica. Su hardware es específico para este tipo de tecnologías, aprovechando las ventajas que los sistemas biológicos poseen, como bajo consumo de energía, y altas prestaciones de procesamiento. Motivo por el cual es interesante poder conseguir alguna de estas ventajas convirtiendo dispositivos convencionales a dispositivos generadores de eventos a través de sencillas modificaciones de hardware o software. En este capítulo se presentan diferentes metodologías de conversión de dispositivos no basados en eventos a dispositivos basados en eventos.

3.1 Codificación en Eventos

Progresos recientes en redes neuronales basadas en eventos para clasificación y control han motivado el incremento de investigaciones y desarrollos en hardware con base neuromórfica, especialmente en el campo de la robótica. Los controladores basados en eventos han tenido éxito en muchas aplicaciones prácticas. A modo de ejemplificar las capacidades de esta tecnología, algunos ejemplos son el control de orientación [49], *visual servoing* [50], predicción de movimiento [51]. Estas aplicaciones son ejemplos de cómo diferentes grupos de investigadores encuentran útil este tipo de desarrollo y así dar a conocer las características diferenciadoras de los dispositivos neuromórficos. Este potencial de desarrollo ha abierto la posibilidad a la diversificación de investigaciones sobre desarrollos hardware y software con base biológica en aplicaciones de otros ambitos. Estos desarrollos están mayormente relacionados con los avances en las redes neuronales por eventos o Spiking Neural Networks (SNN). Las SNN no solo consideran el espacio, sino también el tiempo tal como lo hacen los organismos biológicos. Tanto las entradas como las salidas de las SNN deben estar codificadas en eventos, por lo que los sensores, los actuadores y en consecuencia el controlador debe estar basados en eventos, despertando un interés en no solo crear controladores y algoritmos [52], [53], [54] basados en eventos sino también los sensores y actuadores [55], [56]. En este capítulo se plantean diferentes enfoques de conversión de dispositivos convencionales a dispositivos que generan eventos haciéndolos compatibles con sistemas neuromórficos o simplemente para aprovechar sus cualidades.

En el campo de la robótica, la visión artificial y las unidades de medida inercial (IMU) son cruciales para la navegación y posicionamiento. Usualmente estas unidades consisten en acelerómetro, giroscopio, magnetómetro y cámara de visión. El acelerómetro y giroscopio proporcionan la información de un cuerpo con respecto a un plano, su

movimiento y rotación, mientras que el magnetómetro ayuda a determinar la orientación con respecto a un campo magnético, usualmente la Tierra. Los mecanismos robóticos requieren interactuar con su entorno por lo que la visión artificial es determinante para esta tarea. Estos sensores son los más comúnmente utilizados en la robótica móvil como medio de retroalimentación del entorno de exploración. El conjunto de estos sensores puede requerir un alto nivel de procesamiento, incrementando requerimientos de recursos de procesamiento, almacenamiento y energía. La codificación basada en eventos puede resultar como una posible solución.

Para demostrar que diferentes sensores no basados en eventos pueden ser tratados como tal, y generar información en eventos para poder ser codificada en AER, se han elegido dos diferentes sensores útiles para la navegación de sistemas robóticos: un compás magnético y un sensor de ultrasonidos y por otra parte, una cámara de visión es un sensor muy importante para la interacción con el entorno. Con esto se quiere demostrar la posibilidad de incorporar sistemas no generadores de eventos en sistemas que procesan eventos. Aunque se ha estado haciendo referencia de la importancia de estos elementos en la robótica, su campo de aplicación puede extenderse más allá, como son los sistemas de detección y geolocalización.

En las siguientes secciones se explicarán diferentes metodologías para la codificación de información de sensores convencionales a sensores basados en eventos, por una parte se explicarán las metodologías que se proponen para el sensor de ultrasonidos y el compás, en el que solo hay un canal de salida, y posteriormente la metodología propuesta para la cámara de visión con $M \times N$ canales de salida según el número de píxeles.

3.2 Generación de Eventos y Codificación AER para Sensores de Ultrasonidos y Compás Digital

La generación de eventos y codificación AER de los sensores de ultrasonidos y compás digital se ha planteado desde dos diferentes enfoques: eventos por cambios de magnitud y eventos por valor de magnitud.

3.2.1 Tipos de Generación de Eventos

En el primer enfoque los datos son generados solamente cuando un cambio en la magnitud anterior ha sido medido: un cambio en la posición angular en el compás digital o un cambio en la distancia en el sensor de ultrasonidos. La principal característica que se busca con este enfoque de codificación es lograr una alta reducción de transmisión de datos sin perder la posición relativa en caso del sensor de ultrasonidos y manteniendo la posición absoluta en el compás digital. Todo con un bajo coste computacional.

El segundo enfoque, eventos por valor de magnitud, está constituido para enviar eventos de manera continua sin importar si el valor ha cambiado o no. Este codifica la magnitud enviando periódicamente el tiempo transcurrido desde el último evento generado, lo que resulta en tren de pulsos de codificación de la orientación o la distancia. Utilizando esta codificación es posible mantener la posición absoluta tanto en el compás digital como en el sensor de ultrasonidos en todo momento. Este tipo de codificación es ideal para el uso en sistemas de aprendizaje y entrenamiento de redes neuronales bioinspiradas [57], [58] por su alta producción de datos y su redundancia de información. La codificación propuesta es la que se ve en la ecuación 3.1.

$$R = f\left(\frac{|S|}{S_{max}}\right) + R_{min} \quad (3.1)$$

$$\frac{S_{max}}{\frac{R_{max}}{\Delta t}} \quad (3.2)$$

En la ecuación 3.1 R_{min} es la frecuencia del evento mínima correspondiente a la ausencia de estímulo S con S_{max} como el valor máximo posible a codificar y R es el valor de la frecuencia a codificar. Los eventos negativos son codificados de la misma manera pero en un canal de salida diferente. La precisión de la codificación está limitada por R_{max} que es la frecuencia máxima aceptada por el dispositivo destino y por la resolución temporal del diferencial de tiempo t , de acuerdo a la ecuación 3.2. La codificación en AER se implementa al agregar a la información el tiempo de impresión o timestamp, del momento en el que se origina cada evento. De esta manera es posible además de hacerlo compatible con otros dispositivos, también se puede visualizar en jAER. Esta codificación requiere una rápida transmisión de datos con un determinismo en el bus utilizado. El código de ambas codificaciones está disponible en [59].

3.3 Generación de Eventos y Codificación AER para Cámara Basada en Fotogramas

A diferencia de los sensores anteriores, aquí la información es dispuesta en $M \times N$ salidas, según el tamaño del arreglo de píxeles, además de que cada salida suele estar codificada a su vez en tres canales, que reflejan comúnmente el espacio de color RGB, haciendo la conversión más compleja que en los dos casos anteriores.

Para la generación y codificación de eventos se diseñaron tres metodologías para transformar los datos generados por la cámara a eventos, con la intención de comparar resultados y desempeño. Si bien hay diferencias en la forma de generación del evento, las tres tienen en común la manera de tratar los tres canales RGB que definen a cada

píxel, ya que para las tres metodologías es requerido un valor único de intensidad de la iluminación. Para esto se propone pasar del espacio de color RGB al YCBCR, en donde el valor que nos interesa es Y, que corresponde al valor de luminosidad de la imagen.

Las ecuaciones 3.3 a 3.6 presentan el cambio de espacio de color RGB de 8 bits de resolución a YCBCR y viceversa. Se ha seleccionado el espacio RGB de 8 bits por canal, ya que es el mayormente implementado por las tecnologías basadas en fotogramas, y dominante en aplicaciones de robótica móvil.

$$R = Y + 1,402(CR - 128) \quad (3.3)$$

$$G = Y - 0,34414(CB - 128) - 0,7144(CR - 128) \quad (3.4)$$

$$B = Y + 1,772(CB - 128) \quad (3.5)$$

$$Y = 0,299R + 0,587G + 0,114B \quad (3.6)$$

El valor de Y es el que principalmente se comparará y actualizará para la generación de eventos en las siguientes metodologías propuestas:

- **Metodología 1:** Se hace una primera captura del fondo de la imagen y se guarda en un arreglo M x N, según el tamaño del sensor. Posteriormente se compara píxel a píxel los que existen en el arreglo guardado, con los que provienen de la cámara fotograma a fotograma. Si el valor de la diferencia entre ambos píxeles actual es mayor en magnitud a un valor mínimo definido como umbral, se producirá un evento, en consecuencia, el evento podrá ser negativo o positivo según el valor de la diferencia.

- **Metodología 2:** En la segunda metodología de conversión se propone, al igual que en la metodología 1, hacer una comparación píxel a píxel. Se producirá un evento positivo si la diferencia es mayor en magnitud y sentido al valor de umbral pero además que en la posición $[(X-1), Y]$ y $[X, (Y-1)]$ no se haya superado también el valor de umbral. De la misma manera, se producirá un evento negativo si la diferencia es menor a un valor de umbral negativo y considerando también lo que ha sucedido con la posición $[(X-1), Y]$ y $[X, (Y-1)]$. A diferencia de la metodología 1, el valor del arreglo a comparar se actualiza una vez evaluado cada fotograma con el valor actual de cada posición.
- **Metodología 3:** La tercera metodología de conversión consiste en realizar las mismas operaciones y evaluaciones que con la metodología 2, pero en lugar de hacer la operación para encontrar un valor iluminación se seleccionará solo un canal de los tres posibles RGB, siendo más económica en el número de operaciones a realizar.

La **metodología 1** es sencilla en cuanto a la ejecución pero es sensiblemente la que más datos produce de las tres, pues compara la matriz de píxeles inicial con las siguientes por lo que los cambios que haya en la escena durante el período de grabación serán en su totalidad eventos a generar. En la **metodología 2** La producción de eventos es significativamente menor, pues se buscan los cambios de magnitud y sentido en el valor calculado de Y , tratando de buscar los contornos de los objetos en la escena. La **metodología 3** está pensada en poder disminuir el tiempo de procesado eliminando el cálculo de Y , y de esta manera tener una respuesta más suavizada y continua de la producción de eventos.

3.4 Pruebas

El objetivo es realizar una comparativa del desempeño normal de los sensores seleccionados frente al desempeño con la metodologías de codificación. Para el sensor de ultrasonidos y el compás digital la tarea principal es comparar la cantidad de datos a transmitir a través de un canal AER. Se plantearon dos diferentes pruebas para poder medir de una manera más objetiva que simplemente observando desde una aplicación de visualización como jAER [60]. La primera prueba fue hecha cuando la magnitud a medir es constante a lo largo del tiempo y la segunda fue haciendo fluctuar la magnitud a lo largo del tiempo. Se consideraron los factores externos de temperatura, fijaciones y mecánica para hacer las pruebas lo más similares posibles para lograr una repetibilidad de los resultados. Para obtener los datos, se probaron tres diferentes métodos de transmisión:

1. Forma clásica. Valor medido directamente a la señal producida y convertida por medio del DAC del microcontrolador y transmitida por puerto serie del microcontrolador.
2. Codificación por cambio en la magnitud.
3. Codificación por valor de la magnitud.

3.4.1 *Sensor de Ultrasonidos*

La configuración para las pruebas realizadas se muestra en el esquema de la figura 3.1. El sensor de ultrasonidos y el compás digital se conectaron a un microcontrolador ATMega2560. El microcontrolador se desempeña a manera de un transductor, lee las medidas de los sensores utilizando su interfaz convencional, por medio de entrada analógica, y la transforma en datos basados en eventos. El microcontrolador puede transmitir los eventos por medio de una salida digital, produciendo eventos en alto/bajo con respecto al tiempo, y la otra opción es trans-

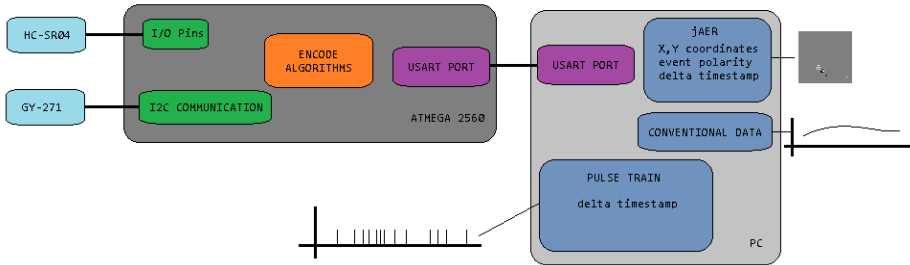


Figura 3.1: Esquema de la implementación del hardware e interconexión de las diversas partes del algoritmo para la codificación de eventos.

mitir los eventos por comunicación a otro dispositivo, encapsulando los datos en alguno de los protocolos AER [61]. Para la visualización y obtención de gráficas se ha utilizado jaER y Matlab, al ser plataformas robustas y probadas para este tipo de metodología de codificación. En la selección de hardware se buscó un sensor fácilmente accesible y de precio económico, probando que prácticamente cualquier sensor puede transformarse a un sensor basado en eventos sin tener que hacer una gran inversión. El sensor ultrasónico fue el HC-SR04. El sensor de ultrasonidos HC-SR04 es capaz de medir una distancia de 2 a 4000 cm con una precisión de hasta 3mm.

Una vez medido un valor desde el sensor, se estima la distancia según el rango medio del sensor y se genera un evento a través del microcontrolador. En el caso del enfoque eventos por cambio de magnitud el valor actual medido es comparado con el anteriormente medido, el valor actual medido es comparado con el anterior, y si es diferente un evento es generado. La figura 3.2 muestra los datos visualizados desde jaER. Este puede ser visto análogamente a cómo el objeto al que está apuntando el sensor de ultrasonidos se aleja o se acerca relativamente del sensor.



Figura 3.2: Representación de los eventos generados por el sensor de ultrasonidos a través del microcontrolador y visualización en jAER. Los píxeles blancos indican que la distancia medida entre el sensor y el objeto se reduce, por el contrario lo píxeles negros indican que la distancia medida comienza a hacerse más grande.

En el caso del enfoque de eventos por valor de la magnitud, cada medida es convertida en un tren de pulsos. Altas frecuencias implican que el sensor mide una distancia corta. Se ha codificado en 4 bytes e implica el tiempo de impresión desde el último evento.

Valor constante

Esta prueba consiste en comparar los datos cuando el valor de la magnitud es constante y no hay un cambio físico real de movimiento relativo entre el sensor y el objeto al que apunta.

La tabla 3.1 muestra los valores medidos a diferentes distancias por un tiempo de 20 segundos.

La codificación por cambio en la magnitud genera menos datos por lo que el uso de ancho de banda para la transmisión se vuelve significativamente menor, dejando espacio para la transmisión de otros datos tanto de este dispositivo, como de alguno otro que pudiera tener acceso a la red. Este resultado puede ser predecible para el lector, pues solo se generan datos si hay cambio en la magnitud, por lo que al no haber un cambio físico en la posición del sensor con respecto al objeto que mide, solo el ruido captado por el sensor serán los datos a leer. Este ruido es producido por la variabilidad de la frecuencia emitida

Valor constante								
Distancia cm	Datos Binarios		Eventos por Cambio			Eventos por Valor		
	Eventos Totales	Tiempo de Transm. μs	Eventos Totales	Tiempo de Transm. μs	Factor de Reducción %	Eventos Totales	Tiempo de Transm. μs	Factor de Reducción %
33	1212	6060	212	1060	82.5	1212	6060	0
25	1247	6235	882	4410	29.2	1247	6235	0
20	1273	6365	599	2995	52.9	1273	6365	0
16	1290	6450	374	1870	71.0	1290	6450	0
11	1312	6560	725	3625	44.7	1312	6560	0

Tabla 3.1: Resultados del sensor de ultrasonido. En esta tabla se pueden comparar resultados comparando la cantidad de datos que produce el sensor en una transmisión convencional de datos binarios con una medición constante a lo largo del tiempo. El tiempo de duración de todas las pruebas fue de 20 segundos.

por el sensor, las vibraciones del montaje y las irregularidades propias del ambiente. Por esto la relación de eventos producidos no es necesariamente proporcional al cambio en magnitud.

Con respecto a la codificación por valor de la magnitud, la cantidad de datos generados son los mismos que con la medida clásica. Esto es debido a que intrínsecamente el método de codificación genera un evento cada vez que se realiza una medida. Sin embargo, la información codificada con este enfoque está lista para ser procesada por un sistema de SNN.

Valor fluctuante

Para esta prueba, el sensor fue posicionado enfrente de un mecanismo que simulaba ser el movimiento de una leva, figura 3.3. La distancia medida tiene un patrón similar a una señal senoidal desde un punto perpendicular al eje de giro. El período de esta señal también podía ser controlado al modificar la velocidad de giro de la leva. El resumen de los resultados obtenidos se muestra en la tabla 3.2.

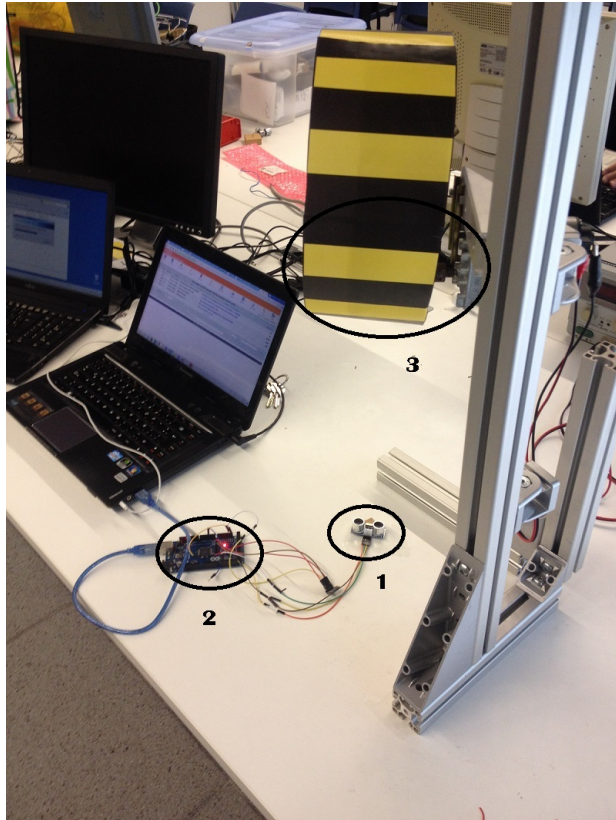


Figura 3.3: En esta imagen se representan el montaje realizado para la medida de distancia para un valor fluctuante con el sensor de ultrasonidos. 1) El sensor de ultrasonidos 2) microcontrolador 3) dispositivo giratorio en forma de leva.

Valor Fluctuante								
Distancia cm	Datos Binarios		Eventos por Cambio			Eventos por Valor		
	Eventos Totales	Tiempo de Transm. μs	Eventos Totales	Tiempo de Transm. μs	Factor de Reducción %	Eventos Totales	Tiempo de Transm. μs	Factor de Reducción %
2.2	1262	6310	1194	5970	5.3	1262	6310	0
2.6	1261	6305	1179	5895	6.5	1261	6305	0
3.5	1261	6305	1175	5875	6.8	1261	6305	0
5.7	1263	6315	1174	5870	7.0	1263	6315	0
15.2	1276	6380	1062	5310	16.7	1276	6380	0

Tabla 3.2: Resultados del sensor de ultrasonido. En esta tabla se pueden comparar resultados comparando la cantidad de datos que produce el sensor en una transmisión convencional de datos binarios tanto con una medición fluctuante a lo largo del tiempo. El tiempo de duración de todas las pruebas fue de 20 segundos.

En este caso es posible observar que los tres métodos tienen un resultado de producción de datos similar, ya que la distancia medida está cambiando.

3.4.2 *Compás Digital*

Al igual que con el sensor de ultrasonidos, el compás digital seleccionado fue uno de fácil disponibilidad, el HMC5883L. Utiliza una magneto-resistencia conectada a un ADC de 12 bits con una frecuencia de actualización máxima de 160 Hz.

Una vez que el valor de orientación es leído desde el compás digital, el arcotangente es calculado de tal manera que el valor del ángulo en radianes nos proporcione la orientación con respecto al campo magnético del norte. Finalmente, los radianes son transformados a grados y ajustados para que den un valor siempre positivo.

El enfoque de codificación por cambio de magnitud fue de la siguiente manera: los grados obtenidos son transformados en una coordenada X/Y, codificada en AER1.0. Cada vez que haya algún cambio se generará un evento en esta coordenada. En la figura 3.4 se puede observar el resultado de un movimiento del sensor sobre su propio eje. La po-

sición del centro superior representa el norte, análogamente como lo hace una rosa de los vientos.

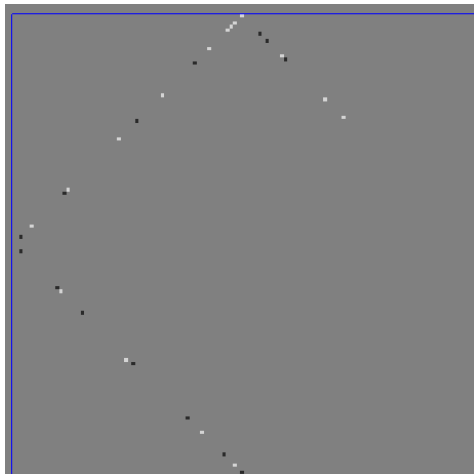


Figura 3.4: En esta figura se representan los eventos generados por el compás digital. La diferencia en el color de los píxeles indica si el sensor gira en sentido horario o antihorario.

Para el enfoque de la generación de eventos de acuerdo al valor de la magnitud, los grados son codificados en un tren de pulsos, con un muestreo fijo y 360 direcciones donde un evento es insertado en la dirección correspondiente según el valor medido. El tiempo de actualización es de 72 ms para así lograr una mejor implementación del algoritmo de codificación, teniendo de esta manera $72/360=0.2\text{ms}=200$ microsegundos, que será el período mínimo implementado por pulso. Se generan N pulsos según la magnitud leída desde el compás.

Valor constante

En estas pruebas se colocó el compás digital en un plano fijo estático paralelo al suelo para así poder medir de manera correcta los grados con respecto al norte polar. Los resultados obtenidos son mostrados en la tabla 3.3.

Valor constante								
Ángulo	Datos Binarios		Eventos por Cambio			Eventos por Valor		
	Eventos Totales	Tiempo de Transm. μs	Eventos Totales	Tiempo de Transm. μs	Factor de Reducción %	Eventos Totales	Tiempo de Transm. μs	Factor de Reducción %
0	2363	11815	116	580	95.0	850680	4253400	\geq bin
90	2363	11815	120	600	94.9	849960	4249800	\geq bin
180	2361	11805	115	575	95.1	850320	4251600	\geq bin
270	2362	11810	116	580	95.0	850680	4253400	\geq bin

Tabla 3.3: Resultados del Compás Digital. Resultados obtenidos al comparar los datos producidos por el sensor al hacer una codificación por cambio de magnitud y valor de magnitud en comparación con los datos binarios que convencionalmente el sensor produce. El período de cada prueba fue de 20 segundos. El factor de reducción en eventos por valor de magnitud es mayor al 100 % debido a que no reduce datos, si no que produce más de los que convencionalmente el sensor produciría si no fuera codificado a eventos. Los datos que aquí se presentan son los generados a un valor constante de medida.

En la tabla 3.3 se puede observar que el factor de reducción de datos del enfoque de codificación basado en el cambio en la magnitud en comparación con los datos con la adquisición clásica es mucho mayor que en el sensor de ultrasonidos, casi del 95.1 %, ya que el sensor es mucho más estable y menos susceptible a las perturbaciones de las condiciones de medición. Por otro lado, el enfoque de codificación según el valor de magnitud es representado por una mayor cantidad de datos, debido a que la codificación del tren de pulsos requiere más datos a transmitir.

Por cada medida tomada puede haber hasta 360 direcciones con su diferencial de tiempo que se envían dependiendo de la orientación, lo cual es muy útil para el entrenamiento de redes neuronales o sistemas con una necesidad de alta transferencia de datos.

Valor Fluctuante								
	Datos binarios		Eventos por Cambio			Eventos por Valor		
Ángulo	Eventos Totales	Tiempo de Transm. μs	Eventos Totales	Tiempo de Transm. μs	Factor de reducción %	Eventos Totales	Tiempo de Transm. μs	Factor de Reducción %
0	2363	11815	208	1040	91.1	4726	23630	\geq bin
90	2362	11810	235	1175	90.0	214942	1074710	\geq bin
180	2362	11810	221	1105	90.6	427522	2137610	\geq bin
270	2362	11810	214	1070	90.9	640102	3200510	\geq bin

Tabla 3.4: Resultados del Compás Digital. Mismas condiciones que en la tabla 3.3 pero bajo una medida fluctuante a lo largo del tiempo.

Valor fluctuante

Para esta prueba el compás fue fijado a un soporte rotatorio que podría girar hasta 270 grados hacia adelante y hacia atrás. Los resultados son mostrados en la tabla 3.4. Como en las pruebas anteriores, la cantidad de datos generada es significativamente reducida por la codificación por cambio en la magnitud, además de que no solo reduce la cantidad de datos producidos, sino que también es más estable que con el sensor de ultrasonidos, lo que provoca mayor reducción de eventos producidos. El enfoque de codificación por el valor de magnitud es similar que el de la adquisición clásica, tal como se podría intuir.

3.4.3 Cámara Basada en Fotogramas

Para las pruebas con una cámara convencional basada en fotogramas, se pensó en dispositivos sencillos y de bajo consumo de potencia, y de esta manera probar la posibilidad de la conversión de fotogramas a eventos en tiempo real. Se consideró que debería poder hacerse la conversión con relativa facilidad y con una inversión en equipos lo más baja posible sin apenas desarrollo de hardware.

Siguiendo la línea de los sensores anteriormente probados, se seleccionó un hardware que pudiese ser sencillamente implementado en aplicaciones de robótica móvil, por tamaño y disponibilidad. El hardware utilizado se describe a continuación:

- Cámara ARDUCAM con chip OV5642, 2592 x 1944 píxeles, 120 fps, 1.4 micrómetros x 1.4 micrómetros, RGB565 y comunicación SPI. Ver figura 3.5.
- Microcontrolador ATMEL AT91SAM3X8E, 512 KB para memoria de programa y reloj de 84 MHz.



Figura 3.5: Cámara con chip OV5642 utilizada para la conversión de fotogramas a eventos [62].

Se debe tener en cuenta que para poder cumplir con un procesamiento con respuesta en tiempo real se deben realizar todas las operaciones de conversión de fotograma a eventos en el menor tiempo posible. Para cumplir con el tiempo más corto posible de procesamiento y transmisión se optó por configurar la cámara con una codificación de tres canales por color por cada píxel codificados en 16 bits, conocido como RGB565. Los primeros 5 bits del primer byte corresponden al canal del color azul, los siguientes 3 bits del primer byte y los primeros 3 bits del segundo byte son el canal verde y finalmente los 5 últimos bits son el canal rojo. Para trabajar con este formato se diseñó una interfaz lógica de transferencia de datos en el microcontrolador para interactuar con la cámara. El microcontrolador debe enviar comandos de configuración antes de poner la cámara operativa, en el que se establece entre otras cosas, el tamaño de la escala a capturar, el espacio de color, y la resolución, principalmente. El microcontrolador ejecuta un algoritmo para la obtención de un valor en escala de luminosidad basándose en la ecuación 3.6 y considerando la resolución RGB565

que obtenemos de la cámara. El valor que representa la luminancia Y está codificado en 8 bits o lo que es lo mismo de 0 a 255 valores posibles. En la ecuación 3.7 se describe el cálculo realizado.

$$Y = 3R + 2G + B \quad (3.7)$$

Aunque la aproximación se tiene que truncar por la resolución que nos ofrece la cámara selecciona con el formato RGB565, los resultados son bastante aceptables, beneficiado por el hecho de que el valor real del píxel en sus tres canales de color se convierte a monocanal, pasando de estar representado de 24 bits a 8 bits. Si bien la cámara fue configurada para producir el tamaño más pequeño disponible, 320 x 240 píxeles, el tiempo de procesamiento se alejaba bastante de tener una respuesta que se pudiera considerar como tiempo real. El valor que se calculó en el que se podía representar una imagen con un campo de visión aceptable y a una velocidad adecuada fue de 50 x 50 píxeles.

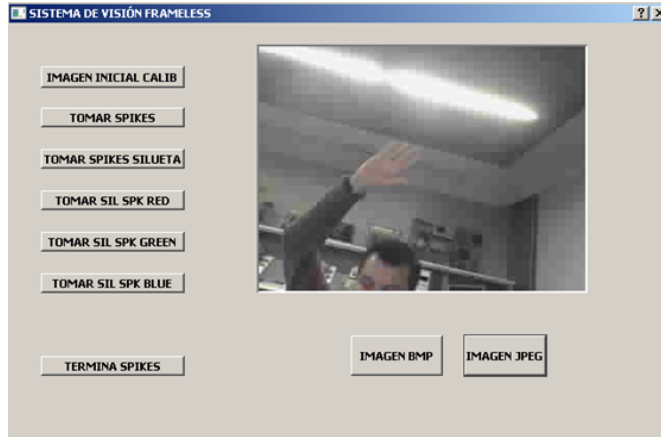


Figura 3.6: Interfaz diseñada en Qt [63] para la configuración, control y operación de la cámara *ARDUCAM*. Con el botón *IMAGEN INICIAL CALIB* se guarda la imagen base a comparar para la metodología 1. Con el botón *TOMAR SPIKES* Se inicia la generación de eventos según la metodología 1. La metodología 2 inicia al dar click en *TOMAR SPIKES SILUETA*. La metodología 3 se activa según el canal que RGB que se quiera seleccionar. También se han creado las opciones para que la cámara capture en los formatos convencionales BMP o JPEG. El botón *TERMINA SPIKES* sirve para dejar de producir eventos independientemente de la metodología seleccionada.

Para la selección de la metodología e interacción con la cámara y microcontrolador por parte del usuario, se utilizó el entorno de diseño Qt [63]. En esta plataforma se diseñó una aplicación a modo de interfaz de usuario para el control y selección del modo de operación de la cámara. La aplicación se muestra en la figura 3.6.

En la figura 3.7 se muestran varias secuencias bajo diferentes escenas y niveles de iluminación de la conversión de fotogramas a eventos visualizadas desde jAER. Se puede comprobar que a pesar de la baja resolución de la cámara con la que se realizaron las pruebas, y la matriz de análisis pequeña (50x50 píxeles) los resultados son satisfactorios pues se alcanzan a distinguir con cierta notoriedad los objetos en la escena.

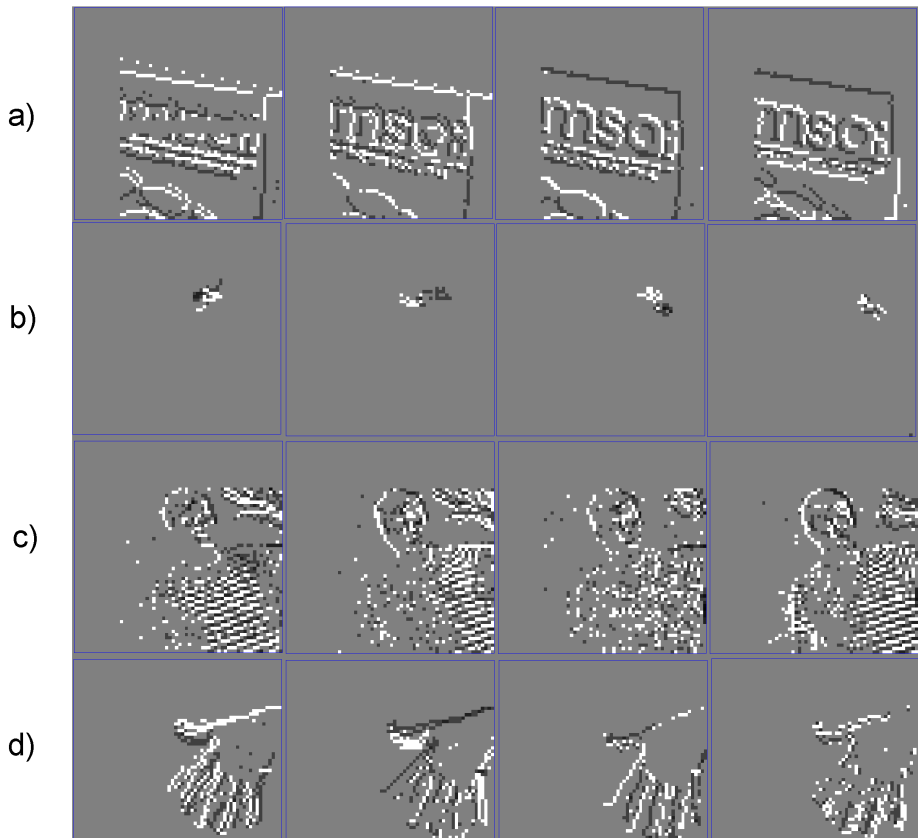


Figura 3.7: Diferentes secuencias de datos AER bajo diferentes condiciones de iluminación y a diferentes velocidades de movimiento. a) Secuencia de un papel con letras impresas. b) Hormiga moviéndose aleatoriamente. c) Persona agitando los brazos d) Mano agitando los dedos.

Considerado los resultados iniciales se procedió a realizar una comparativa en el resultado obtenido de las tres metodologías bajo una misma escena controlada. Se puso un disco de color blanco con un cuadro de color rojo en un extremo a girar justo enfrente de la cámara tal como en la figura 3.8. El resultado de esta prueba se refleja en la figura 3.9.

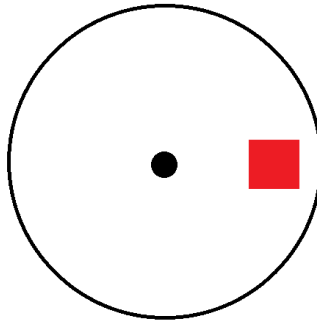


Figura 3.8: Esquema del disco utilizado para la generación de eventos con la cámara basada en fotogramas, girando a 60 rpms.

En la figura 3.9 se presentan 4 secuencias para las tres metodologías. En esta figura se observa lo que a priori podía intuirse según el propósito de cada metodología: la producción de eventos se decrementa notoriamente de la metodología 1 a la 3. En primera instancia se puede apreciar un claro desfase entre el valor anterior, mostrándose de pasar de píxeles negativo en color negro, a píxeles positivos en blanco, efecto generado por el retraso del procesamiento de un fotograma a otro, la codificación y transmisión de eventos, y este retraso es mayor conforme más eventos son generados, causando un mayor tiempo de procesamiento y de transmisión. También se puede observar que para la secuencia de M1, el cuadro es mucho más notorio en la imagen, a diferencia de M2 y M3 que solo se aprecia el contorno. Por otro lado, una diferencia importante entre M2 y M3 es que en M3 se elimina casi en su totalidad los eventos que pudieran ser considerados como

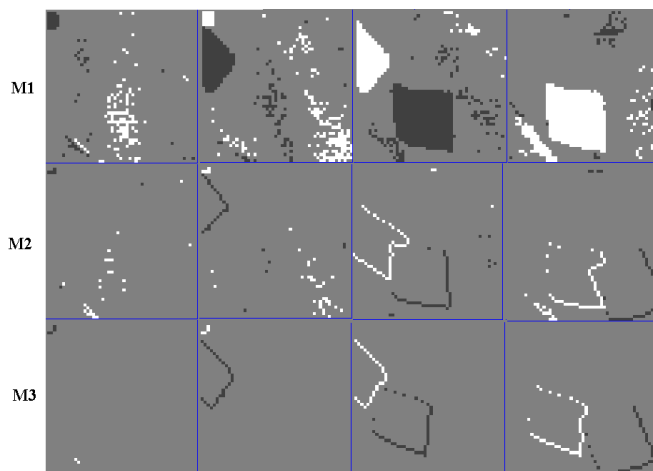


Figura 3.9: Resultado de las tres metodologías propuestas. Secuencia mostrada de un cuarto de giro del disco, obtenida desde jAER. M1) representa la primera metodología, M2) representa la segunda metodología y M3) la tercera metodología con canal R seleccionado.

ruido, por lo que el tiempo de procesamiento y transmisión de eventos se beneficia y se reduce aún más.

3.5 Resumen

Según los resultados obtenidos se puede corroborar que con los diferentes enfoques de codificación es factible transformar un sensor no basado en eventos, de una forma sencilla, en uno basado en eventos. Además, de que según las necesidades de la aplicación final, es posible transformar y generar eventos en mayor o menor medida.

En términos generales, para los sensores probados, se puede decir que el enfoque de codificación por cambio en la magnitud reduce significativamente la tasa de transferencia tanto si no hay cambios físicos del ente que se está midiendo como si lo hay. No obstante, la calidad de la estabilidad del sensor y su repetibilidad influyen en el comporta-

miento de la reducción de estos datos, característica a considerar si se desean tratar estos datos con un procesamiento de filtrado posterior a la adquisición.

El enfoque de codificación por valor de la magnitud propone una codificación sin reducción de datos de transferencia. Por el contrario, es de utilidad cuando existe la necesidad de datos para sistemas de aprendizaje, donde es más importante la redundancia de datos para una misma información que la posible saturación de la comunicación entre el transductor conversor de eventos y sistemas periféricos de control. Los resultados aplicados al sensor de ultrasonidos y al compás digital nos sugieren que podrían ser fácilmente ampliables a otros sensores de condiciones similares.

Las metodologías propuestas para la conversión de una cámara basada en fotogramas en eventos, obtiene resultados estables y satisfactorios con los matices que no se deben olvidar, como el tiempo de procesamiento y transmisión. Este tiempo incluye al menos la captura del total de la matriz, el envío de petición de fotograma, la transmisión del fotograma, la decodificación del paquete recibido y el procesamiento de la aplicación. Sin olvidar el espacio de memoria requerido y tecnología de procesamiento del sistema conversor. Se ha logrado tener un desempeño aceptable para aplicaciones de tiempo real donde es prioritaria la localización de objetos en movimiento y la definición de los objetos en escena tiene un papel secundario.

Las pruebas presentadas en este capítulo demuestran que utilizar las técnicas de codificación basadas en eventos requiere un bajo coste computacional para la codificación de dispositivos no basados en eventos, además de que provee datos listos para poder ser procesados en sistemas de aprendizaje como las SNN y otros sistemas neuromórficos. El resultado obtenido nos permite poder mezclar sensores de visión, posicionamiento y orientación, todas en conjunto en un sistema neuromórfico que puede aumentar sus posibilidades de toma de decisión y

control, diversificando y ampliando el panorama de dispositivos disponibles que generan datos en forma de eventos. Todo esto manteniendo un bajo coste computacional y con requerimientos simples de hardware.

Capítulo 4

Algoritmos de Procesado de Eventos

Los sistemas neuromórficos basados en eventos han tenido poco auge más allá de los laboratorios, ejemplo de esto es que los sistemas de visión con esta tecnología no se han incorporado en la industria de la automatización industrial a pesar de contar con cualidades aprovechables en el sector. En este capítulo se expone un algoritmo de filtrado de eventos con el fin de preparar la información para aplicaciones que requieren poca tasa de transmisión en tiempo real. Se propone el algoritmo bio-inspirado LDSI Less Data Same Information. Este algoritmo tiene el objetivo de localizar los eventos que pueden ser considerados como ruido y descartarlos, además de disminuir el efecto estela que este tipo de tecnologías generan. Por otra lado, se exponen dos nuevos algoritmos para obtener una aproximación numérica y objetiva del nivel de ruido en diferentes conjuntos de datos de eventos, llamado ENM Events Noise Measurement y del nivel de parecido entre un conjunto de datos sin filtrar y después de ser filtrado, llamado ESM Edge Similitude Measurement.

4.1 Punto de Partida

En ambientes donde la respuesta en tiempo en real y la simplicidad de los datos a procesar son requisitos vitales para el buen desempeño de un sistema, el desempeño depende de la cantidad de datos en red y la disponibilidad de la misma para transportar información. En la industria, dos ejemplos típicos de estas necesidades son los sistemas de control de ejes o control de movimiento, *visual servoing* [64] y los sistemas de inspección visual por ordenador. Los sistemas de control de movimiento necesitan una alta disponibilidad de red y los sistemas de visión requieren una alta tasa de transmisión de datos y ambos una respuesta en tiempo real. Estos tiempos de respuesta suelen estar por debajo de los 10 milisegundos y cada vez es más común encontrar tiempos cercanos o inclusive por debajo del milisegundo.

Los sistemas de visión artificial basados en eventos bio-inspirados generan eventos solo en las posiciones del arreglo de píxeles en donde la carga de intensidad de luz varía, por lo que cada sensor puede producir un evento de manera continua sin necesidad de un *disparador*, y solo dependiente de un nivel umbral. Este hecho reduce la necesidad de transmitir y procesar datos redundantes en un arreglo de $M \times N$ píxeles como sucede con los sistemas de visión tradicional. Sin embargo, la variabilidad de la luz ambiente y el movimiento de objetos alrededor de la zona de inspección entre otros factores, pueden inducir variantes en la efectividad de esta tecnología. Tal motivo ha hecho que se haya planteado la formulación de un algoritmo con diferentes parámetros de filtrado para el procesamiento de los datos producidos por las cámaras basadas en eventos. La finalidad de que sea parametrizable es para filtrar de manera más estricta o más permisiva la cantidad de eventos a transmitir dependiendo de las condiciones de la escena y de la aplicación.

El ajuste de los parámetros del algoritmo es un paso importante pues el desempeño dependerá de lo que se quiere evaluar y procesar en la es-

cena capturada. Por este motivo es interesante obtener una aproximación a manera de índice numérico de la cantidad de datos considerados como ruido y de la semejanza del resultado después del filtrado con el original, y de esta manera poder evaluar objetivamente el desempeño de los ajustes del algoritmo de filtrado LDSI.

4.2 Algoritmo LDSI

El algoritmo planteado para el filtrado de datos de eventos ha sido llamado LDSI (*Less Data Same Information*) o Menos Datos Misma Información. Con el nombre se puede entrever que el objetivo es reducir los datos a transmitir dentro de una red de dispositivos pero manteniendo la información relevante. La reducción de datos trae consigo una reducción en el hardware necesario para el procesamiento y por ende una reducción de consumo de energía y requerimientos de almacenamiento de memoria.

El algoritmo LDSI ha sido diseñado y estructurado para una fácil configuración de los parámetros de filtrado según las necesidades y condiciones de cada aplicación final, ya que los niveles de ruido o datos indeseados muchas veces dependen de factores no posibles de controlar. Concretamente, en el caso de las cámaras basadas en eventos, pueden ser cambios de iluminación repentinos, fuentes de luz intermitentes, sombras o reflejos, entre otros.

Si bien el algoritmo tiene una base neuromórfica, el algoritmo no trata de emular el funcionamiento biológico de una neurona, sino extraer algunas ventajas que pueden ser útiles a la hora de la implementación hardware y software. El modelo definido y construido está representado en la figura 4.1.

El diseño está basado en capas para el procesamiento de los eventos entrantes desde una cámara basada en eventos con una estructura análoga a la estructura de una neurona. Está organizado de la siguien-

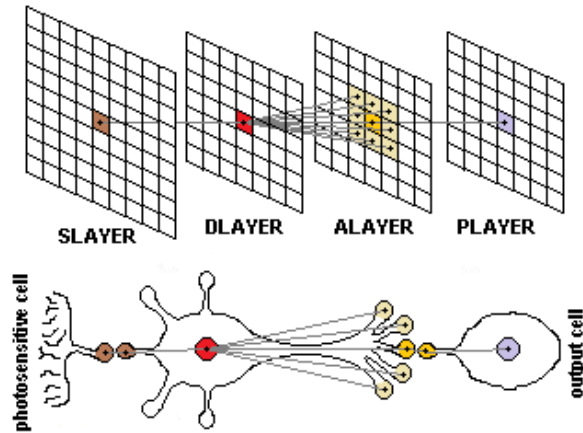


Figura 4.1: Interrelación entre las diferentes capas de unidades del algoritmo LDSI.

te manera: *SLAYER*, unidades del sensor, en este caso, los píxeles de la cámara, que actúa como las dendritas alimentando al núcleo *DLAYER* que a su vez reenvía el estímulo de información a la capa *ALAYER* como lo hacen las terminales sinápticas. Cada terminal sináptica en *ALAYER* produce una salida que transfiere el evento a *PLAYER*, que pudiera ser considerado como *DLAYER* en sucesivas cadenas de neuronas. Para facilidad del lector, se aconseja establecer una relación del nombre de las capas con su función por lo que se puede asociar *SLAYER* como capa del sensor, *DLAYER* capa de las dendritas, *ALAYER* como la capa de los axones y *PLAYER* como la capa de impresión o salida. El modelo del algoritmo LDSI define una neurona compuesta de dos unidades asociadas, por un lado el núcleo o *DLAYER* y el axón con sus terminales sinápticas o *ALAYER*. Cada unidad en *DLAYER* y *ALAYER* recibe eventos desde la capa *SLAYER*, y estos eventos modifican el potencial interno de la unidad a la que llegan.

Una unidad en *DLAYER* recibe entradas de eventos desde una fuente *SLAYER* externa pero localizada en la misma posición XY, simultáneamente, *ALAYER* recibe un estímulo en la misma posición XY

pero además en sus vecindades. Cada unidad en cada capa modifica su potencial con la llegada de los estímulos anteriormente descritos y cuando supera un valor de umbral se refleja un evento en *PLAYER*. Este enfoque permite que las unidades de *PLAYER* puedan ser utilizadas para postprocesamiento aplicando algoritmos de tratamiento de la información o reconocimiento de la misma. Este funcionamiento es análogo a los sistemas de procesamiento tradicionales donde se aplica un preprocesado para filtrar y eliminar elementos no deseados y posteriormente aplicarles algoritmos de detección, clasificación y cualquier extracción de información.

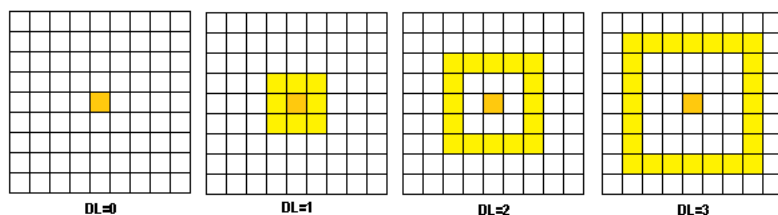


Figura 4.2: Interconexiones entre *ALAYER* y *DLAYER* dependiendo del parámetro *DL*.

El número de unidades vecinas de *DLAYER* que estimulan a la capa *ALAYER* está definida por un parámetro *DepthLevel* (*DL*), como se muestra en la figura 4.2. El algoritmo, como anteriormente se ha mencionado, está pensado para ser parametrizado y poderse adecuar a cambios de intensidad en la iluminación de la escena, movimientos rápidos de objetos, etc, según la aplicación final. Los parámetros a ajustar son los siguientes:

Nivel de excitación en *DLAYER* (*ELD*): Es la magnitud de potencial que incrementa la unidad en la posición *XY* de *DLAYER* cuando un evento es recibido desde la misma coordenada *XY* pero desde la capa *SLAYER*.

Nivel de excitación en ALAYER (ELA): Es la magnitud de potencial que incrementa la unidad en la posición XY de *ALAYER* cuando un evento es recibido desde la misma coordenada XY desde la capa *DLAYER*.

Nivel de excitación en las unidades vecinas de ALAYER (ELAN): Cuando aparece un evento en una coordenada XY en la capa *DLAYER*, ELAN corresponde al incremento del potencial en las unidades vecinas a la de la coordenada XY de *ALAYER* y definido por el parámetro DL, anteriormente descrito.

Nivel de umbral en DLAYER (TPD): Definen el mínimo valor de excitación en una unidad de *DLAYER* para producir un evento de salida.

Nivel de umbral en ALAYER (TPA): Define el mínimo valor de potencial requerido en *ALAYER* para producir un evento de salida.

Decremento de potencial en DLAYER (DPD): El valor de potencial a decrementar en *ALAYER* una vez que ha pasado un tiempo definido por *MTR*.

Decremento de potencial en ALAYER (DPA): El valor de potencial a decrementar en *ALAYER* una vez que ha pasado un tiempo definido por *MTR*.

Además de los parámetros anteriores, existen otras variables que se ven afectadas por el tiempo de impresión de los eventos de llegada:

Tiempo actual de impresión(AT): El tiempo de impresión del evento actual de salida de *SLAYER*.

Tiempo de la última impresión (LT): El tiempo de impresión del último evento procesado por el algoritmo.

Diferencia de tiempo (DT): Diferencia de tiempo entre el evento que actualmente se procesa con el último procesado.

Máximo tiempo a recordar (MTR): Es el valor a comparar con DT, si este es menor, los potenciales de *DLAYER* y *ALAYER* de todas las unidades se decrementarán según los parámetros correspondientes.

$$\vartheta_D(x, y)_{t+1} = \begin{cases} \vartheta_D(x, y)_t + ELD, & \text{Si evento en } D(x, y) \\ \vartheta_D(x, y)_t - DPD, & DT \geq MTR \\ 0, & (\vartheta_D(x, y)_t \geq TPD) \\ & \text{AND } (\vartheta_A(x, y)_t \geq TPA) \\ 0, & (\vartheta_D(x, y)_t - DPD) \leq 0 \\ \vartheta_D(x, y)_t, & \text{sin evento AND } DT < MTR \end{cases} \quad (4.1)$$

$$\vartheta_A(x, y)_{t+1} = \begin{cases} \vartheta_A(x, y)_t + ELA, & \text{si evento en } D(x, y) \\ \vartheta_A(x, y)_t + ELAN, & \text{si evento en DL} \\ & \text{vecindad de } D(x, y) \\ & \text{AND } DT < MTR \\ \vartheta_A(x, y)_t - DPA, & DT \geq MTR \\ 1, & (\vartheta_D(x, y)_t \geq TPD) \\ & \text{AND } (\vartheta_A(x, y)_t \geq TPA) \\ 0, & \vartheta_A(x, y)_t = 1 \\ 0, & (\vartheta_A(x, y)_t - DPA) \leq 0 \\ \vartheta_A(x, y)_t, & \text{sin evento AND } DT < MTR \end{cases} \quad (4.2)$$

La interrelación de los diferentes parámetros en la capa *ALAYER* y *DLAYER*, se presentan las ecuaciones 4.1 y 4.2. Una vez que llega un evento en la posición XY de *SLAYER*, el potencial de *DLAYER* en la posición XY, además en un nivel DL de vecinos en *ALAYER*, el potencial es incrementado según su correspondiente nivel de excitación. La ecuación 4.1 muestra como el potencial ϑ en la posición XY de *DLAYER* cambia y en la ecuación 4.2 muestra la evolución del poten-

cial en una unidad XY de la capa ALAYER, conforme los parámetros previamente definidos.

Algoritmo 1 Algoritmo LDSI: procesando los eventos que se generan en SLAYER.

Entrada Datos en forma de eventos de la capa SLAYER

Salida Eventos en la capa PLAYER

```

1: if SlayerINPUT_EVENT in (x,y) then
2:   {- Incremento de potencial}
3:    $DT = AT - LT$ 
4:    $D_{(x,y)} = D_{(x,y)} + ELD$ 
5:    $A_{(x,y)} = A_{(x,y)} + ELA$ 
6:   for  $i = -DL$  to  $DL$  do
7:     for  $j = -DL$  to  $DL$  do
8:        $A_{(x+i,y+j)} = A_{(x+i,y+j)} + ELAN$ 
9:     end for
10:  end for
11:  if  $DT < MTR$  then
12:    {- Decremento de potencial}
13:    for all  $i = 0$  to  $((M - 2) * DL)$  do
14:      for all  $j = 0$  to  $((N - 2) * DL)$  do
15:        if  $D_{(i,j)} \geq DPD$  then
16:           $D_{(i,j)} = D_{(i,j)} - DPD$ 
17:        else
18:           $D_{(i,j)} = 0$ 
19:        end if
20:        if  $A_{(i,j)} \geq DPA$  then
21:           $A_{(i,j)} = A_{(i,j)} - DPA$ 
22:        else
23:           $A_{(i,j)} = 0$ 
24:        end if
25:      end for
26:    end for
27:  end if
28:  if  $D_{(x,y)} \geq TPD$  AND  $A_{(x,y)} \geq TPA$  then
29:     $P_{(x,y)} = EVENT$ 
30:     $D_{(x,y)} = 0$ 
31:     $A_{(x,y)} = 0$ 
32:  else
33:     $P_{(x,y)} = NULL$ 
34:  end if
35:   $LT = AT$ 
36: end if

```

Como es posible analizar en el algoritmo 1, si se reciben eventos en cualquier unidad XY, el potencial es incrementado en dicha unidad. En caso de DLAYER, es incrementado solo cuando la coordenada es exactamente la misma que en SLAYER. Para ALAYER, el potencial es incrementado en la posición XY además de las unidades circundantes definidas por el parámetro DL. El potencial de todas las unidades es decrementado después de que durante un tiempo mayor a MTR no se

haya recibido ningún estímulo de *SLAYER*. Cuando las condiciones de cada unidad sobrepasan los niveles de umbral establecidos para cada capa se genera un evento y el potencial de dicha unidad se reinicia. En la figura 4.3 se puede observar un ejemplo del comportamiento descrito por el algoritmo.

En esta figura se muestra el comportamiento de las distintas capas del algoritmo LDSI: en caso de llegada de eventos en dos unidades diferentes de la capa *SLAYER*, XY y $X(Y+1)$, el valor del potencial en cada una de las unidades se incrementa respectivamente según los valores parametrizados. Cuando este potencial es superior a un nivel de umbral entonces es generado un evento de salida. También es posible observar que si durante un tiempo establecido no hay ningún evento de entrada el potencial de las unidades *ALAYER* y *DLAYER* es decrementado pudiendo alcanzar como mínimo el valor 0. En caso de *DLAYER*, solo se ve influenciada por los eventos generados por la unidad XY de *SLAYER*, sin embargo, las unidades de *ALAYER* son afectadas tanto por el evento en la posición XY como por todos los eventos de las unidades circundantes hasta el nivel parametrizado.

El modelo propuesto en este algoritmo recuerda el funcionamiento a grandes rasgos del cerebro de un mamífero, recibiendo constantemente estímulos y dependiendo de las conexiones y de la diferencia de tiempo entre los diferentes estímulos genera una salida a una capa superior de neuronas, la cual también será en forma de estímulo.

4.2.1 Verificación del Algoritmo LDSI

Todos los parámetros anteriormente descritos se han configurado como valores enteros positivos, y experimentalmente se ha observado que, para tener un desempeño alto en un hardware sencillo, se deberían mantener los valores de ELD, ELA, ELAN, TPD, TPA, DPD y DPA entre 0 y 10; de lo contrario, el coste computacional se incrementa,

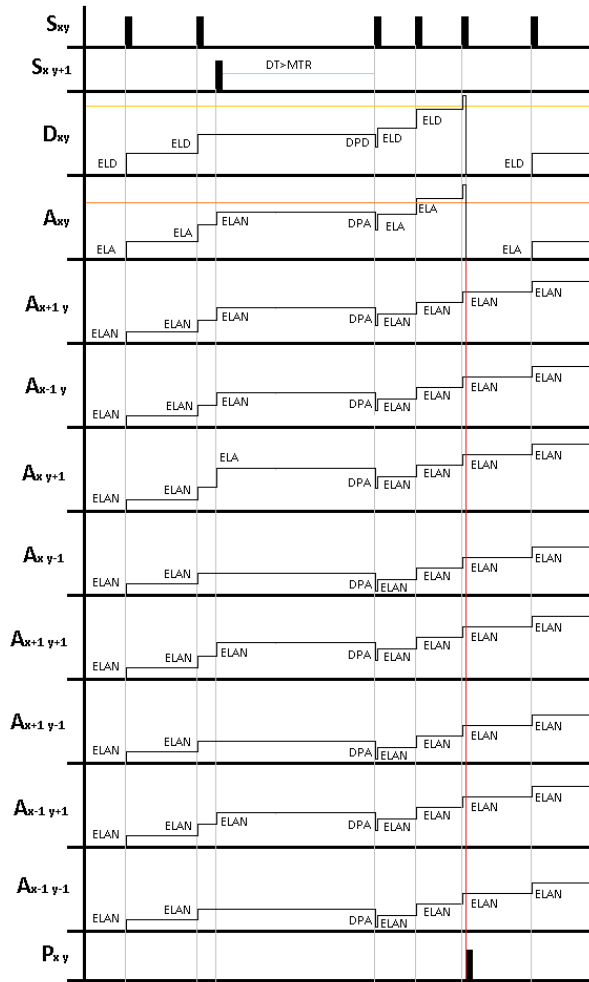


Figura 4.3: Ejemplo gráfico en caso de la llegada de eventos en dos unidades diferentes de la capa SLAYER, XY y $X(Y+1)$. El valor del potencial en cada una de la unidades se incrementa con la llegada de eventos, y cuando este potencial es superior a un nivel de umbral, un evento de salida es generado. Si durante un tiempo parametrizado no hay ningún evento de entrada, el potencial de las unidades ALAYER y DLAYER es decrementado. En caso de DLAYER, solo se ve influenciada por los eventos generados por la unidad XY de SLAYER, sin embargo las unidades de ALAYER se ven influenciadas por la unidad en esta posición más las circundantes a ella.

lo que conlleva a sacrificar mayor consumo de energía y tiempo de procesamiento.

La implementación del algoritmo LDSI fue diseñada en dos etapas: en la primera etapa el algoritmo fue probado de manera *offline*, mediante ficheros de datos previamente grabados con cámaras basadas en eventos y codificando el algoritmo en lenguaje C++ se analizó el resultado obtenido por el algoritmo. La segunda etapa fue de manera *online* que fue implementar el algoritmo en una FPGA y visualizar el resultado. Se realizó enviando los datos AER a la FPGA y visualizándolos en tiempo real en la aplicación jAER como en las pruebas del capítulo anterior.

4.2.2 Descripción del Hardware

Para desarrollar este algoritmo se optó por utilizar plataformas y herramientas software y hardware comunes en el procesamiento de los datos generados por dispositivos basados en eventos. jAER [60] es la aplicación elegida como medio de visualización de los resultados a manera de interfaz con el usuario. El algoritmo se verificó de manera *offline*, mediante archivos previamente grabados por cámaras de visión basadas en eventos que eran filtrados por código escrito en C++.

Una vez verificado que el algoritmo cumplía con su cometido de manera *offline*, se procedió a ejecutarlo y verificarlo de manera *online*. Se implementó el código en una FPGA que comunicaba con una cámara basada en eventos, el esquema es el que se muestra en la figura 4.4. La FPGA usada fue la VIRTEX 6 XC6VLX240T-1FFG1156 presente en la ML605 evaluation board de Xilinx.

La cámara conectada fue una DVS por puerto paralelo de INILABS [65]. El puerto paralelo de esta cámara utiliza un bus AER de siete pines para la coordenada del eje Y, siete para la coordenada del eje X, uno para la polaridad, y finalmente dos señales del control de co-

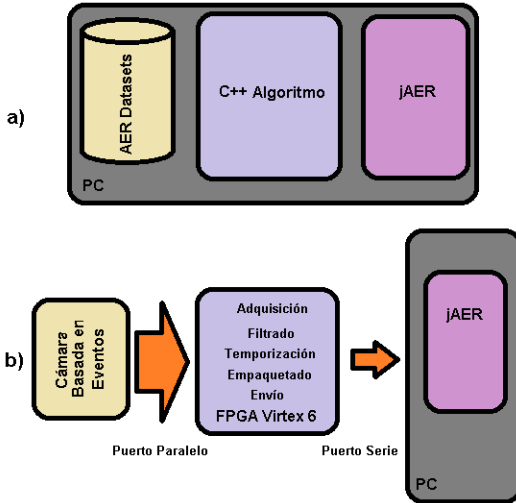


Figura 4.4: Esquema de desarrollo del algoritmo LDSI a) muestra la estructura de la implementación *offline* y en b) la implementación *online*.

municación *REQ* y *ACK*. Por otro lado, la FPGA fue conectada por puerto serie a un PC para poder visualizar el resultado del filtrado. El baudrate utilizado fue 921600 bps con el protocolo AER 2.0 para la interpretación de los datos por jAER a través de un bus serie desde la FPGA.

Prueba OFFLINE

Para la ejecución del algoritmo de manera *offline*, se utilizaron las diferentes bases de datos que han desarrollado grupos de investigación en el campo [66]. El algoritmo fue aplicado a diferentes ficheros de datos grabados por cámara basadas en eventos con diferentes tipos de escenas, cantidad de ruido y velocidad de movimiento en la escena, con el propósito de observar su comportamiento con diferentes magnitudes de ruido y cantidad de eventos por unidad de tiempo. El objetivo de lo anterior es poder encontrar algún tipo de interrelación entre los parámetros y así definir reglas de comportamiento del algoritmo según las aplicaciones y necesidades.

Prueba ONLINE

En la ejecución *online* del algoritmo LDSI, se migró la codificación de C++ a VHDL para poder incorporarlo en una FPGA pensando en optimizar de esta manera la velocidad de operación, el almacenamiento de memoria y la paralelización de operaciones. Las pruebas se realizaron en una FPGA Virtex 6 (evaluation kit), utilizando las diferentes entradas y salidas de propósito general para la interconexión de los datos AER. La salida del algoritmo se canalizó por medio de una comunicación UART a un ordenador dónde poder visualizarlos en tiempo real.

La implementación del hardware fue hecha en una FPGA XILINX VIRTEX6 6VLX240TLFF1156-11[67]. El sistema desarrollado incluía una interfaz serie para comunicación con un PC donde se ejecutaría la aplicación jAER, para así grabar los diferentes resultados. La ocupación lógica total para la comunicación, el algoritmo y el empaquetado AER fue de 671 LUT lo que es un valor bajo en este tipo de tecnologías. No fue necesario utilizar ningún elemento adicionales al reloj externo de la tarjeta, que para este caso era de 50 MHz.

4.2.3 Resultados del Algoritmo LDSI

Como primer paso se analizó una escena de un objeto moviéndose enfrente de la cámara basada en eventos. Se hizo una comparación visual a través de la aplicación jAER para comprobar en un primer momento el comportamiento del filtro.

En la figura 4.5 se puede observar que el filtro no solo tiene un comportamiento adecuado para definir mejor las siluetas y que limpia los píxeles fácilmente reconocidos como ruido, sino que también reduce la cantidad de datos, pasando de 2633 eventos a 1329 eventos concretamente para este ejemplo, lo equivale a una reducción del 50 %. Se

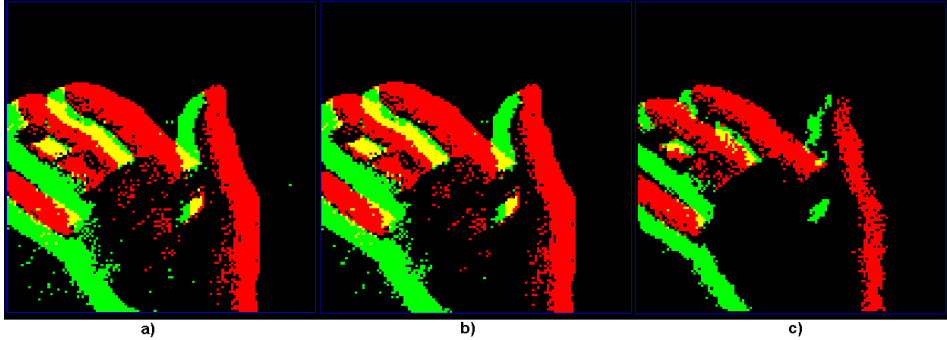


Figura 4.5: Comparativa de diferentes resultados de aplicación del algoritmo LDSI. a) Eventos originales. b) Eventos producidos después de aplicar el algoritmo LDSI con parámetros poco restrictivos. c) Resultado de los eventos producidos con parámetros restrictivos.

puede apreciar que permanecen los eventos con mayor importancia de la información de la escena.

Con el objetivo de obtener una completa evaluación del algoritmo LDSI, se creó un conjunto de diferentes combinaciones de valores para todos los parámetros del algoritmo LDSI, llegando a sumar más de 100 000 combinaciones posibles, número que se consideró adecuado y representativo del comportamiento y desempeño del algoritmo, para poder extraer mejores conclusiones. El resultado del desempeño del algoritmo de todas estas combinaciones fue analizado con el fin de encontrar un patrón de comportamiento según el ajuste y combinación de los parámetros.

El rango de resultados obtenidos por el algoritmo es muy amplio, varían desde una alta reducción de ruido hasta una alta reducción de eventos. Sin embargo, es necesario considerar que, a más alta reducción de eventos mayor es la reducción no solo de ruido sino también de la definición de los eventos que contienen información útil de la escena, usualmente borde o siluetas. Por otra parte, existen ciertas combinaciones de parámetros con poca acción de filtrado, que hacen compor-

tarse el algoritmo como un repetidor, pues básicamente los datos que entran son los mismos que se obtienen como salida. En cualquier caso, el algoritmo, nunca genera más eventos que los que se reciben ni tampoco deforma la escena. En cambio, la puede reducir hasta el grado de no producir ni un solo evento de salida. Este hecho es necesario tenerlo en cuenta en el momento de seleccionar parámetros en el filtro según la aplicación final, es decir, para aplicaciones en donde se requiere una baja tasa de uso de banda de comunicaciones, reducción de hardware de procesamiento y reducción de consumo de energía. El algoritmo tiene un comportamiento ideal, sin embargo, en aplicaciones donde se requiere acentuar el peso de cada evento, como pasa en las redes neuronales, el desempeño del algoritmo puede ser medio, y en aplicaciones donde se requiere producir eventos como complemento de los que realmente generó la cámara, el desempeño del algoritmo es bajo.

De la figura 4.6 a la figura 4.10 se muestra el desempeño del algoritmo en relación de los parámetros con el coeficiente de reducción de datos. Por obvias razones, las combinaciones que generan una reducción de 100 % de datos filtrados, es decir, cero eventos de salida, han sido descartados. Se han seleccionado estas figuras entre las múltiples combinaciones de parámetros, ya que representan de una mejor manera la tendencia de comportamiento.

En la figura 4.6 se muestra el número de eventos de salida generados por el algoritmo LDSI, en función de los parámetros TPD y ELD mientras se mantienen constantes el resto de parámetros. Si el valor de ELD se acerca a TPD por la izquierda, la probabilidad de evento generado se incrementa, por lo que se han descartado las combinaciones en donde ELD es mayor que TPD, pues no aportaban información trascendental al ser iguales la salida y la entrada. El análisis del resultado de la comparación de estos dos parámetros es fácilmente distinguible en la figura 4.7, ya que a menor valor de TPD, mayor cantidad de eventos serán generados mientras se incrementa el valor de ELD y se

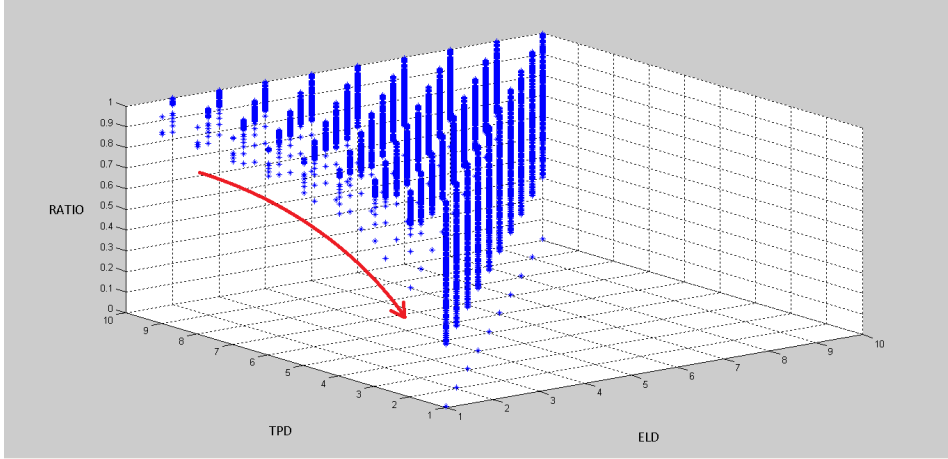


Figura 4.6: Coeficiente de reducción de datos (Ratio) en función de TPD y ELD. Se puede observar una clara tendencia en la que a mayor TPD con ELD constante, la ratio se acerca a uno, lo que se significa que la reducción se acerca a casi el 100 % de datos eliminados.

mantengan los demás parámetros constantes. En la figura 4.7 se observa una tendencia en la que a mayor diferencia entre ELA y TPA, siendo este último el mayor, se incrementará el coeficiente de reducción de eventos, por lo que el filtro será más restrictivo en la generación de eventos. Por otra parte, la relación $TPA/ELAN$ mostrada en la figura 4.9 muestra también que la tendencia de reducción de datos se incrementa conforme la diferencia entre TPA y ELAN se incrementa, relación también importante al parametrizar qué tan estricto debería ser el filtro para generar eventos. Finalmente, en la figura 4.10 se muestra una relación menos clara que las anteriores, pues no es fácilmente distinguible una tendencia, aunque se puede observar ligeramente que a mayores valores de DPD y TPD el filtro se hace más restrictivo, pero el margen es mayor que con otras tendencias.

A pesar de que la relación entre los parámetros no es lineal, después de decenas de miles de ejecuciones del algoritmo, se pueden notar ciertas tendencias de comportamiento, en el que bajo condiciones similares

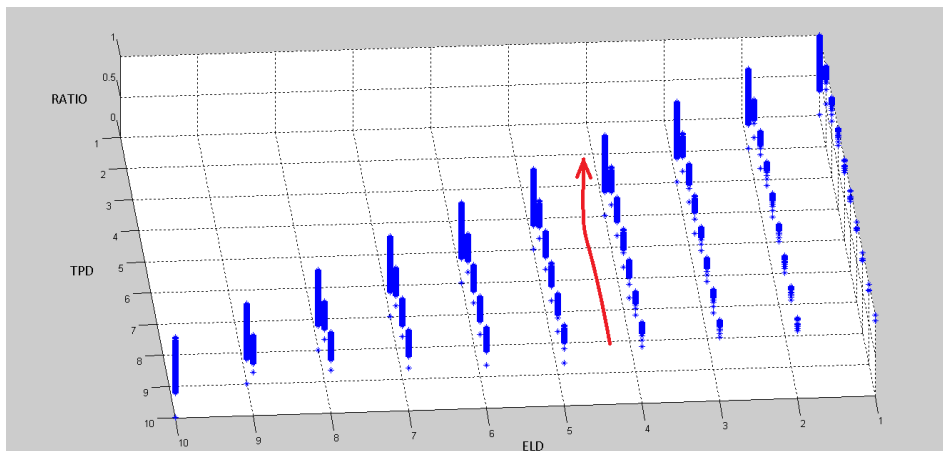


Figura 4.7: Relación TPD/ELD. Muestra una relación más clara en el algoritmo. La reducción de eventos sube de forma casi exponencial el acercarse ELD a TPD.

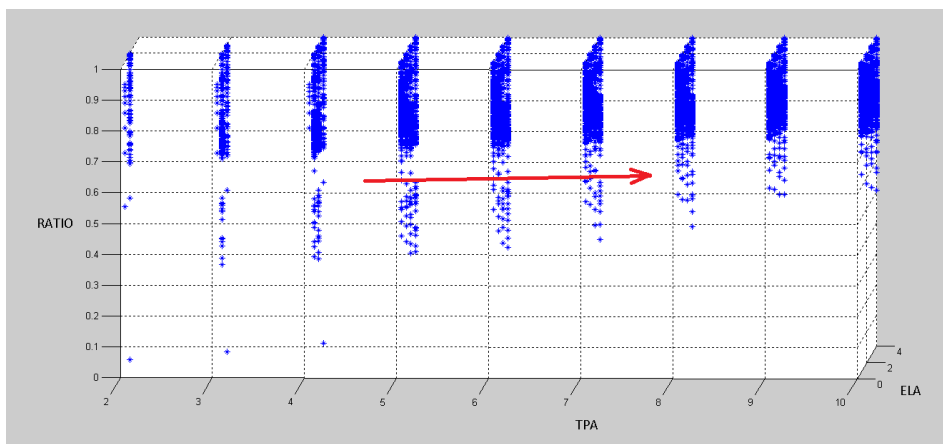


Figura 4.8: Relación TPA/ELA. Un valor de TPA mayor que de ELA reduce la cantidad de eventos producidos, lo que elimina una gran cantidad de ruido, pero también puede eliminar información relevante de la escena.

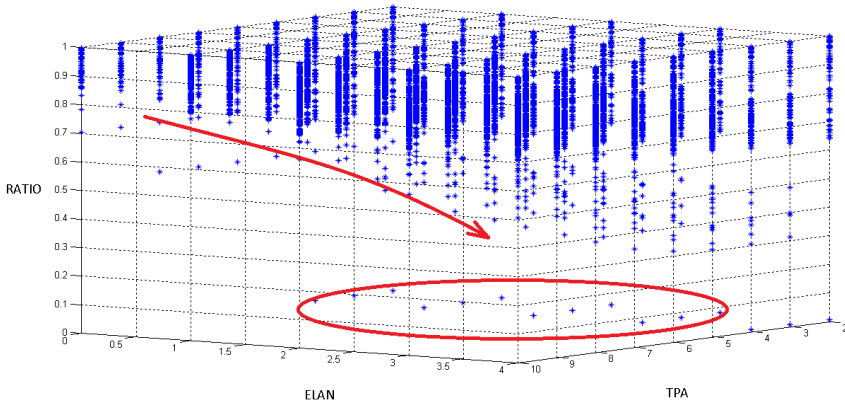


Figura 4.9: Relación TPA/ELAN. A mayores valores de ELAN mayor será el coeficiente de reducción, y esta ratio se ve aún más decrementada cuando TPA es mayor. También se señalizan en un círculo algunos casos en los que el comportamiento puede variar y no seguir la tendencia, pues la parametrización del algoritmo no tiene una respuesta lineal.

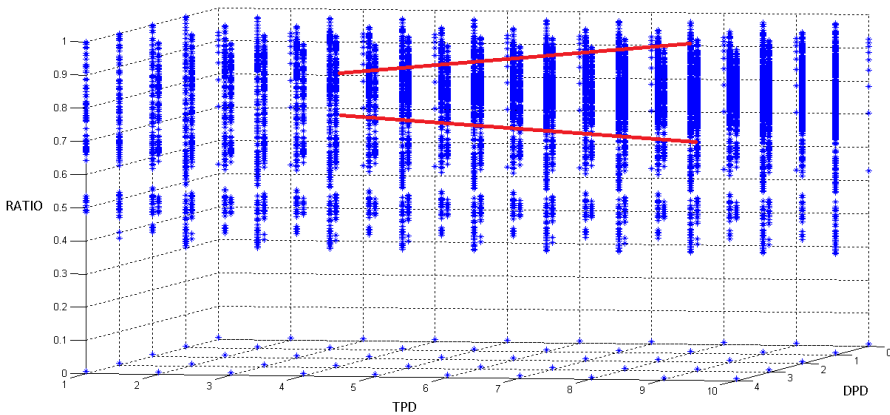


Figura 4.10: Relación TPD/DPD. Es menos clara que las anteriores. Sin embargo, a mayor TPD y mayor DPD se puede observar una tendencia a tener una mayor ratio. Este hecho es predecible ya que el potencial de cada unidad en la capa D-LAYER se decrementa en mayor medida y tiene un mayor nivel de umbral, dificultando así la producción de eventos de salida.

los resultados también son similares. Por lo que se pueden realizar las siguientes observaciones:

1.- Para aplicaciones donde se tiene una alta fuente de ruido en relación con los eventos que realmente se quieren obtener como información útil de la escena: el algoritmo LDSI tiene un comportamiento mejor con valores menores del parámetro MTR. Lo que implica que continuamente se estén decrementando los valores de las capas *DLAYER* y *ALAYER*, manteniéndose cargadas aquellas unidades de cada capa que mayor actividad tienen y descartando a las de actividad esporádica. Una fuente de ruido típica en estas cámaras es una lámpara de luz incandescente.

2.- Cuando la cantidad de ruido en una escena es casi equiparable a la cantidad de eventos que pueden producirse en la escena, es recomendable definir valores pequeños de ELA y valores de ELAN ligeramente mayores, con TPA mayor que ambos.

3.- Si la eliminación del ruido no es el objetivo principal, sino la distinción de las siluetas de los objetos en movimiento en la escena, es recomendable tener valores de ELD altos con DPD y DPA similares.

4.- Es importante eliminar parámetros donde los resultados pueden resultar obvios y predecibles tales como $ELD=ELA=0$ que no producirían ningún evento, y por lo contrario, valores de TPD y TPA menores que ELD y ELA respectivamente, pues convertirían al algoritmo en un repetidor de eventos.

Los puntos anteriores no son reglas matemáticas, ya que dependen de diversos factores y múltiples variables tales como la relación entre la cantidad de ruido en la escena con respecto a los datos reales de información, la velocidad de los objetos moviéndose en escena, el tamaño del sensor, y cambios de intensidad repentina de iluminación.

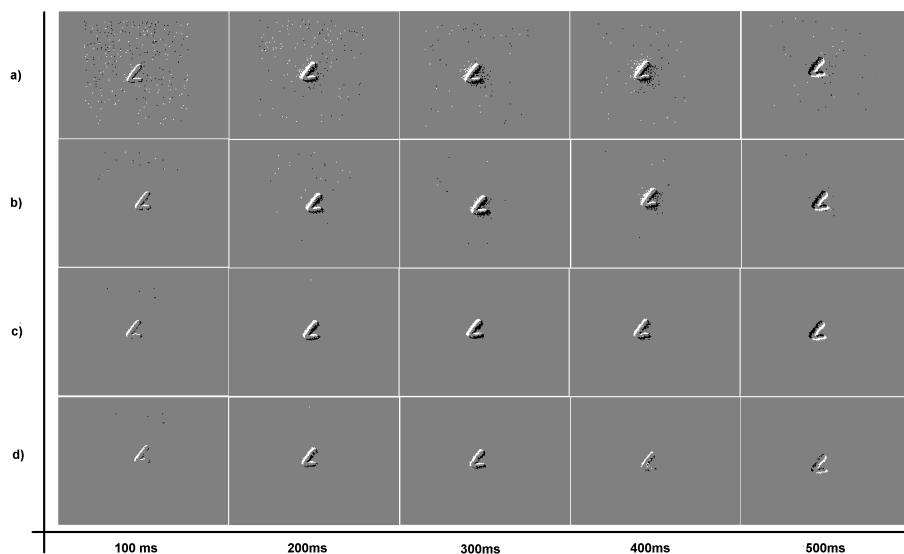


Figura 4.11: Imágenes de la secuencia transcurrida en 500 ms (100 ms por imagen) para una cámara de eventos, capturada desde jAER. a) Datos originales sin tratamiento del algoritmo LDSI b) Baja reducción de los datos y débil eliminación de ruido c) Reducción de datos media y media eliminación de ruido d) Alta reducción de datos y también alta eliminación de ruido.

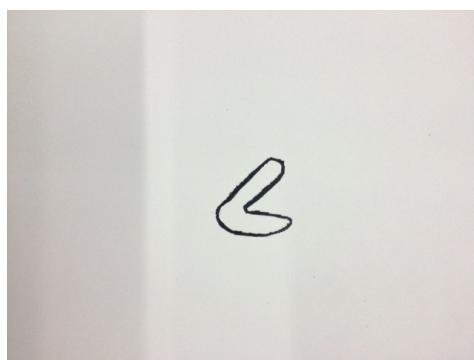


Figura 4.12: Imagen original de la letra *L* escrita a mano y captada por una cámara convencional, basada en *frames* o fotogramas.

En la figura 4.11 se muestra el resultado del desempeño del algoritmo LDSI parametrizado de tres formas diferentes: baja restricción, medianamente restrictivo y alta restricción de generación de eventos, en un transcurso de 500ms de la misma escena grabada con una cámara de eventos. Se han seleccionado 5 instantes cada uno de 100ms del movimiento de una letra *L* escrita a mano como se muestra en la figura 4.12.

Dentro de la figura 4.11, la secuencia en la fila a) representa los datos originales, la fila b) representa el resultado del algoritmo LDSI ajustando los parámetros de una manera poco restrictiva. El resultado arroja un 33 % de reducción de eventos generados, con respecto a los eventos originales generados por la cámara. En la fila c) el resultado reduce casi el 50 % de los eventos ajustando los parámetros a un nivel medio de restricción de generación de eventos. Finalmente, la columna d) muestra el resultado de aplicar a los datos de la escena al algoritmo con un nivel restrictivo alto, produciendo una reducción de casi el 85 % de generación de eventos. Para demostrar la eficacia del algoritmo, además de los resultados visuales se analizó el porcentaje de similitud entre los datos sin filtrar producidos por una cámara basada en eventos y la imagen real escrita a mano. Se utilizó un software especializado en la inspección industrial, Sherlock Visión System de la empresa DALSA[68].

La mayoría de los sistemas industriales cuentan con un algoritmo llamado reconocimiento de patrones o reconocimiento de similitudes con el que se puede comparar un conjunto de imágenes a otras. Para las pruebas realizadas, se enseñó al sistema de visión la imagen real de la letra **L** escrita mano y después se pasaron las secuencias obtenidas desde jAER. El resultado es el que se muestra en la figura 4.13.

La figura 4.13 no solo muestra que el algoritmo LDSI es bueno al eliminar el ruido en esta secuencia de imágenes, sino que también puede facilitar el reconocimiento de los objetos, alcanzando porcentajes de

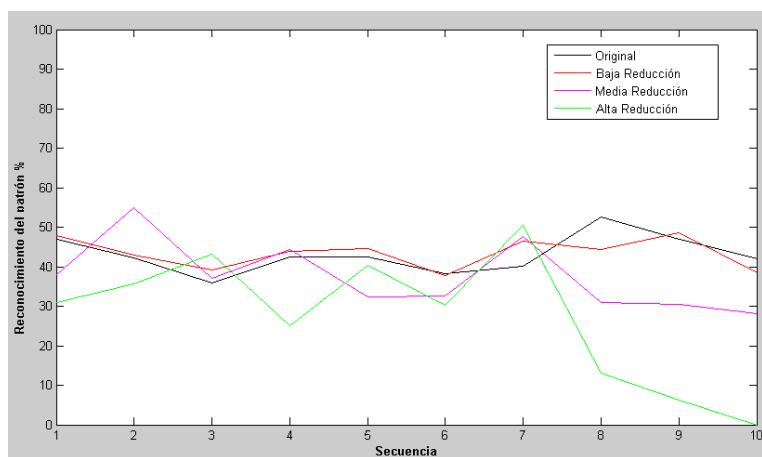


Figura 4.13: Resultado del algoritmo de reconocimiento de patrones. Se muestra el resultado tratando de encontrar el patrón en la secuencia capturada mediante una cámara basada en eventos. Se compara con el resultado del algoritmo de reconocimiento de patrones utilizando el resultado del algoritmo LDSI con diferentes niveles de restricción de producción de datos: débil, medio y alto.

similitud iguales o mayores que el de la imagen real sin filtrar. Este resultado es coherente con el hecho de que el algoritmo mejora la definición de los contornos de los objetos que se mueven en la escena. Es importante que el lector tenga en consideración que los porcentajes de similitud son relativamente bajos debido a que se están obteniendo de una imagen producida por una cámara basada en eventos, lo que transforma sustancialmente la imagen original, captada por una cámara basada en fotogramas.

La tabla 4.1 muestra la reducción de eventos producidos por el algoritmo, considerando que por encima de un 85% de reducción de datos la secuencia resultante es difícil de reconocer por el software de inspección. No obstante, con una reducción del 33%, el resultado de similitud era satisfactorio al ser muy parecido al resultado de la imagen original.

Muestras	Tamaño (bytes)	Reducción %	Reconocimiento %	Desviación Estándar
Secuencia Original	153576	-	43.02	4.01
Filtro Débil	103176	32.82	43.04	3.68
Filtro Medio	51696	66.34	37.64	8.92
Filtro Restrictivo	22614	85.28	27.5	16.52

Tabla 4.1: Resumen comparativo entre las diferentes parametrizaciones del algoritmo LDSI y del reconocimiento del resultado de cada parametrización por medio del software de visión artificial Sherlock.

Como resultado, se plantean los siguientes análisis:

- 1) La configuración del algoritmo con parámetros poco restrictivos puede reducirnos más de un 30 % el número de eventos reales, mejorando la definición de las siluetas de los objetos en escena y reduciendo el ruido alrededor de las mismas.
- 2) La configuración con parámetros altamente restrictivos solo produce el 15 % de los eventos reales, y dependiendo de la velocidad del objeto en escena puede aún contar con información relevante.
- 3) El rango de similitud a lo largo de la secuencia obtenida desde datos de una cámara basada en eventos y traducida en fotogramas para su análisis en un software de inspección, ronda entre el 33 - 66 % muy similar al obtenido a la secuencia original sin la aplicación del filtro.

Hasta aquí se han expuesto los resultados obtenidos en cuanto a reducción de datos conteniendo la misma información, siendo satisfactorios. La evaluación se ha hecho utilizando métodos comparativos enfocados a imágenes obtenidas en cámaras basadas en fotogramas. En la siguiente sección se expone un algoritmo que proporciona una aproximación numérica contando el número de eventos que en un conjunto de datos puede considerarse como ruido y obtener una relación con respecto al conjunto de datos que se puede considerar, *a priori*, como parte del objeto en la escena.

4.3 Metodología de comparación

Para poder realizar una comparativa de una manera más objetiva en cuanto a los resultados obtenidos con el algoritmo LDSI, se diseñó una metodología para obtener una aproximación del número de eventos considerados como ruido y la similitud que guarda el resultado con el conjunto de eventos originales. La metodología consiste en dos algoritmos para el conteo de eventos candidatos a ser considerados como ruido, a los que se les ha llamado ENM de *Events Noise Measurement* para el conteo del ruido, y por otro lado el algoritmo que nos dé una aproximación de qué tan similar sigue siendo el resultado del algoritmo al original, a este se le ha llamado ESM *Edge Similitude Measurement*, el cual aprovecha la capacidad inherente de esta tecnología de definir bordes o siluetas.

Aproximar el cálculo del ruido en un conjunto de datos es importante a la hora de hacer sintonizaciones de los parámetros de distintos filtros y a su vez para poder tener un parámetro o índice numérico de la calidad de la medición, importante si se desea realizar distintas comparaciones sobre un mismo banco de datos, o bien, cuando las condiciones de la adquisición de datos son variantes y no se pueden controlar. Un ejemplo es el caso de una cámara basada en eventos que capta de manera indeseada las irregularidades de la luz ambiente por la obstrucción de la fuente de luz hacia la zona donde la cámara está captando los datos, como puede ser el paso de una nube, una sombra, la frecuencia y tecnología del alumbrado, entre otros factores.

En contraparte, la detección de ruido en imágenes obtenidas por cámaras convencionales basadas en fotogramas tiene un respaldo muy amplio y con una gran variedad de algoritmos y formulaciones de detección y tratamiento de imagen [69], [70], [71], [72], [73], con bibliografía muy extensa, pero que está fuera del ámbito de este trabajo. Sin embargo, hay pocos trabajos en lo que respecta al cálculo del ruido con el objetivo de tener un valor numérico fácil de contrastar; ejemplos

de estos son [74], [75], [76] y [77]. Considerando que la literatura para el tratamiento y procesamiento de imágenes convencionales es muy superior a la que se puede encontrar basada en eventos, los trabajos realizados en este ámbito sobre el cálculo de ruido no son numerosos. Sin embargo, existen algunos trabajos que consideran la importancia de tratamiento de conjuntos de datos basados en eventos en sistemas de visión tales como [78], [79] y [80].

La metodología de comparación propuesta en este trabajo para la aproximación de la cantidad de datos que pueden ser considerados como ruido en un conjunto de eventos y el índice de similitud entre diferentes conjuntos de datos basados en eventos es el resultado de la convergencia de dos algoritmos diferentes, en donde por una parte se busca la cantidad de eventos producidos que tienen pocos o nulos eventos circundantes, tanto en tiempo como en espacio, y por otro lado, la estrategia que saca ventaja del hecho que las cámaras basadas en eventos suelen encontrar los bordes de los objetos en escena, y comparar de esta manera las longitudes de dichos bordes. Esto se realiza calculando la longitud de las siluetas que forman los eventos activos en un mismo periodo de tiempo.

Por términos de practicidad y disponibilidad de datos, los test y resultados se realizaron con datos producidos por cámaras basadas en eventos, sin embargo, el algoritmo podría ser trasladado a cualquier dispositivo generador de eventos en el que puedan ser representados en un plano XY. Las pruebas se realizaron tanto con bases de datos disponibles [66] como con cámaras reales basadas en eventos.

4.4 Algoritmo de Medición del Ruido en Eventos, ENM

En primer lugar, se localizan aquellos eventos candidatos a considerarse como ruido en un período de tiempo establecido y parametrizable. Esto se logra mediante la creación de dos capas o matrices paralelas que activan o desactivan distintas direcciones según la dirección activada por el evento. En la figura 4.14 podemos observar las capas creadas. En la capa del lado izquierdo, denominada REC, de color morado se representa el evento real de entrada producido por una cámara basada en eventos. En la capa del lado derecho, denominada NEI, que hemos denominado capa de relación, se presentan todas las direcciones que se verían influenciadas por los eventos de la capa izquierda dado el algoritmo 2 y 3 que a pesar que forman parte de un mismo algoritmo se ha dividido en dos etapas, por terminos de legibilidad para el lector.

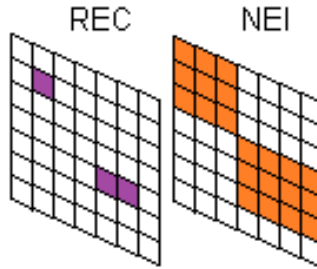


Figura 4.14: La capa izquierda representa los eventos de entrada al algoritmo en el plano XY que son generados desde un dispositivo basado en eventos, en la capa de la derecha se muestran todos los eventos que serán activados en relación a la dirección del evento de entrada. En los algoritmos 2 y 3 son representadas estas capas por **REC** y **NEI**, respectivamente.

Previo a la ejecución del algoritmo ENM (Events Noise Measurement) se debe definir un tiempo máximo de evaluación que será la máxima diferencia que podrá existir entre el primer evento y el último evento de un conjunto de datos a analizar. La primera instancia del algoritmo es activar en la primera capa *REC* las direcciones del conjunto de datos seleccionado. Seguidamente se activan en la capa *NEI* todas

las posiciones circundantes al evento de la primera capa. En la segunda etapa se realiza un primer barrido de conteo evento por evento del conjunto de datos que ocurrieron en el lapso de tiempo ajustado, contabilizando el número de direcciones activadas en la segunda capa alrededor de la dirección activada en la primera capa. Se realiza un segundo barrido de conteo a una mayor distancia que el primer barrido en el que se vuelven a contabilizar el número de direcciones activadas, pero solo los de la capa de entrada. Con el número de eventos activos obtenido de cada barrido para ambos conteos, se decide si el evento es un candidato a ser considerado como evento de ruido. Si el conteo de los barridos anteriores es menor a un parámetro S , que se refiere a la sensibilidad de ruido que se desea detectar, entonces el evento se considera como ruido y se contabiliza como tal. Conforme se hace el barrido se desactivan los eventos analizados, las direcciones inspeccionadas se desactivan, tanto para el primer barrido como para el segundo barrido. Al terminar con un borde se busca el siguiente evento activo e inicia nuevamente el proceso con un nuevo conjunto de datos en el mismo lapso de tiempo predefinido.

Algoritmo 2 Primera etapa del algoritmo ENM. Aproximación del conteo de eventos que pueden ser considerados como ruido.

Entrada Datos en forma de eventos en un plano XY

Salida Eventos en la capa **Nei** y **Rec**

```

1: if INPUT_EVENT en (x,y) then
2:   Rec(x,y) = true
3:   Nei(x,y) = true
4:   for a = 0to8 do
5:     if a>0 y (a<4) then
6:       Py = y + 1
7:     else if a>4 y (a<8) then
8:       Py = y - 1
9:     else
10:      Py = y
11:    end if
12:    if a=2 y (a=6) then
13:      Px = x
14:    else if a>4 y (a<8) then
15:      Px = x - 1
16:    else
17:      Px = x
18:    end if
19:    Nei(Px,Py) = true
20:  end for
21: end if

```

Algoritmo 3 Segunda etapa del algoritmo ENM. Aproximación del conteo de eventos que pueden ser considerados

como ruido.

Entrada Eventos en la capa **Nei** y **Rec**

Salida Posición del evento considerado como ruido

```

1: for a = 0to8 do
2:   if a > 0y(a < 4) then
3:     Py = y + 1 + AB
4:   else if a > 4 y (a < 8) then
5:     Py = y - 1 - AB
6:   else
7:     Py = y
8:   end if
9:   if a = 2 y (a = 6) then
10:    Px = x
11:   else if a > 4 y (a < 8) then
12:    Px = x - 1 - AB
13:   else
14:    Px = x + 1 + AB
15:   end if
16:   if Nei(Px,Py) o Rec(x,y) then
17:     Mascara1 = Mascara1 + 1
18:     Nei(Px,Py) = 0
19:   end if
20: end for
21: for a = 0to16 do
22:   if a = 0o(a = 8) then
23:     Py = y
24:   else if a = 1 o (a = 7) then
25:     Py = y + 1 + AB2
26:   else if a = 9 o (a = 15) then
27:     Py = y - 1 - AB2
28:   else if a > 1 y (a < 7) then
29:     Py = y + 2 + AB2
30:   else
31:     Py = y - 2 - AB2
32:   end if
33:   if a = 4o(a = 12) then
34:     Px = x
35:   else if a = 5 o (a = 11) then
36:     Px = x - 1 - AB2
37:   else if a = 3 o (a = 13) then
38:     Px = x + 1 + AB2
39:   else if a > 5 y (a < 11) then
40:     Px = x - 2 - AB2
41:   else
42:     Px = x + 2 + AB2
43:   end if
44:   if Nei(Px,Py) o Rec(x,y) then
45:     Mascara2 = Mascara2 + 1
46:     Nei(Px,Py) = 0
47:   end if
48: end for
49: if (Mascara1 + Mascara2) < S then
50:   El INPUT_EVENTen(x, y) es un evento considerado como ruido
51: end if

```

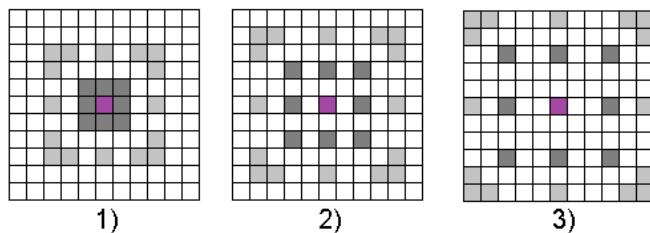


Figura 4.15: Representación del campo de acceso del algoritmo ENM con diversas configuraciones de acción, 1) parámetros $AB1=0$ y $AB2=1$, 2) parámetros $AB1=1$ y $AB2=2$ y finalmente en 3) parámetros $AB1=2$ y $AB2=3$.

En la figura 4.15 se representan 3 posibles configuraciones del algoritmo ENM. Como es posible observar en el algoritmo 3, los parámetros $AB1$ y $AB2$, reflejan la distancia circundante a la que se evaluará el evento de entrada. Aunque si bien es parametrizable, el mejor comportamiento se obtiene con $AB1=1$ y $AB2=2$, puesto que abarca de manera más uniforme las posiciones circundantes de la dirección del evento de entrada.

En la figura 4.16 se observa la ejecución de algoritmo. Del lado izquierdo está la capa de eventos de entrada *REC*, y del lado derecho la capa de relación *NEI*. Se puede también observar que el campo de acción detecta hasta cuatro unidades vecinas para un evento con unidades circundantes activadas, pero un evento más aislado no.

Es importante considerar que la sensibilidad de la detección de posibles candidatos de ruido se verá influenciada por el tiempo máximo de evaluación, siendo recomendable tiempos cortos para grandes flujos de información y tiempos largos para movimientos ocasionales. Además, es necesario tener en cuenta la resolución del dispositivo de eventos, de importancia en el caso de una cámara basada en eventos que trata de captar algo fuera de su profundidad de campo.

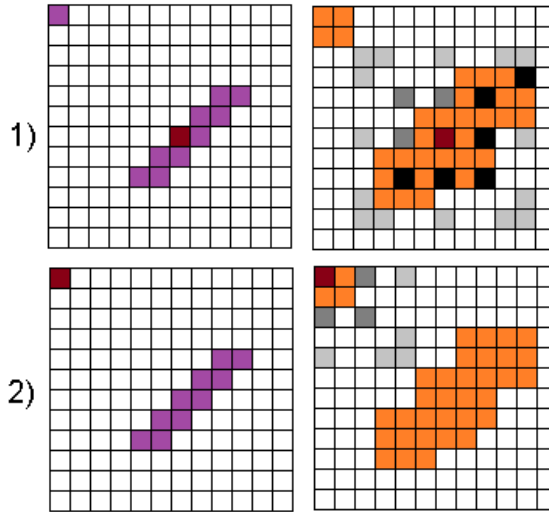


Figura 4.16: En esta figura se muestra la acción de algoritmo ENM en donde en 1) se evalúa un evento de entrada que tiene varias unidades vecinas activas y en 2) un evento que no tiene ningún vecino activo en el campo de acción del algoritmo, considerando este último como un evento de ruido. De color morado están representados los eventos de entrada desde el sensor. En color marrón se señalizan los puntos a evaluar. En gris se localizan las coordenadas de la matriz que se inspeccionarán, cambiando la tonalidad según el nivel de cercanía, y finalmente en negro se marcan los puntos que se han inspeccionado y que están activos, dados los eventos de entrada del sensor.

4.5 Algoritmo de Comparación de Bordes, ESM

Para poder discriminar en mejor medida la cantidad de datos que se pueden considerar como datos del contorno de algún objeto en la escena, se propone una estrategia de contabilización de las direcciones activas contiguas, en un período de tiempo determinado. Este cálculo se basa en obtener el número de direcciones contiguas que se han activado dentro de un tiempo determinado, dando como resultado un valor numérico que por términos prácticos se puede considerar como la longitud de las direcciones contiguas. Posteriormente se calcula

un promedio de las longitudes encontradas. De esta manera se logra discriminar aquellas direcciones activas con longitudes bajas, con la posibilidad de considerarlas como ruido.

Al igual que con el algoritmo ENM, este se dividió en dos etapas. La primera etapa está plasmada en el algoritmo 4 y la segunda en el algoritmo 5. Es necesario definir un tiempo máximo de evaluación que será la máxima diferencia que podrá existir entre el primer evento y el último evento de un conjunto de datos a analizar. Se crea una capa de direcciones del tamaño $2N \times 2M$ donde N, M representan el tamaño de la capa original. En la primera etapa, cada dirección (X,Y) que se active en la capa original activará las direcciones circundantes que se ven en el algoritmo 4. En la segunda etapa, se realiza un barrido en la capa $2N \times 2M$ en el que al igual que la anterior estrategia se contabiliza, pero ahora no de manera estática, sino en el total de las posiciones activadas en la nueva capa, y desactivando conforme se pasa por ellas tal como se muestra en la figura 4.17. El barrido continuo hasta volver a la posición inicial o que no existan más posiciones activas contiguas. El proceso de conteo se repite hasta desactivar el total de las posiciones activadas, promediando el valor numérico obtenido a lo largo del lapso de tiempo seleccionado.

Finalmente, se obtienen dos valores, el número de siluetas o bordes encontrados en un tiempo determinado y la máxima longitud de siluetas encontradas, dentro de este mismo tiempo.

4.6 Resultados de los algoritmos ENM y ESM

Para la implementación de los algoritmos ENM y ESM se utilizó el resultado del algoritmo LDSI, como parte complementaria. De esta manera se podría comprobar de manera objetiva y con el desarrollo de una metodología novedosa la eficacia del algoritmo LDSI en su nivel de filtrado.

Algoritmo 4 Primera etapa del algoritmo ESM. Aproximación de la medición de contornos en un conjunto de datos AER.

Entrada Datos en forma de eventos en un plano XY

Salida Eventos activados en la matriz de Contorno

```

1: while DELTA < CPR do
2:   if INPUT_EVENT in (x,y) then
3:      $X_a = 2x$ 
4:      $Y_a = 2y$ 
5:      $Contorno(X_a, Y_a) = D(X_a, Y_a) + 1$ 
6:      $Contorno(X_a+1, Y_a) = D(X_a+1, Y_a) + 1$ 
7:      $Contorno(X_a+2, Y_a) = D(X_a+2, Y_a) + 1$ 
8:      $Y_a = Y_a + 1$ 
9:      $Contorno(X_a, Y_a) = D(X_a, Y_a) + 1$ 
10:     $Contorno(X_a, Y_a+2) = D(X_a, Y_a+2) + 1$ 
11:     $Y_a = Y_a + 1$ 
12:     $Contorno(X_a, Y_a) = D(X_a, Y_a) + 1$ 
13:     $Contorno(X_a+1, Y_a) = D(X_a+1, Y_a) + 1$ 
14:     $Contorno(X_a+2, Y_a) = D(X_a+2, Y_a) + 1$ 
15:  end if
16:  DELTA = Tiempo(x, y) -Tiempo evento anterior
17: end while

```

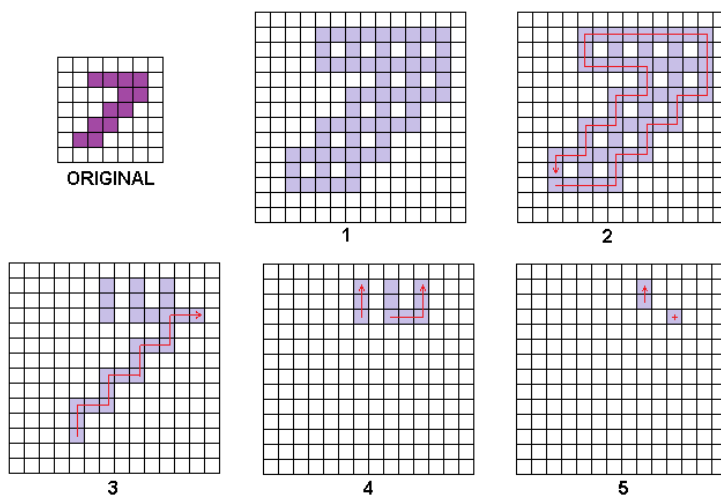


Figura 4.17: Ejemplo de cómo actúa el algoritmo ESM en la medición de contorno. En 1) se activan las posiciones según los datos originales, en 2) se comienza a hacer el barrido dinámico, desactivando aquellas posiciones que se han analizado, 3) hace un nuevo barrido con las posiciones restantes y finalmente en 4) y 5) se hacen barridos hasta desactivar la totalidad de las posiciones.

Algoritmo 5 Segunda etapa del algoritmo ESM. Aproximación de la medición de contornos en un conjunto de datos

AER.

Entrada Datos en forma de eventos en en la matriz de Contorno

Salida Promedio de la longitud del borde en escena

```

1:  $Pox = P_{sx}$ 
2:  $Poy = P_{sy}$ 
3: while La posición  $Pox, Poy$  sea diferente al punto de partida do
4:    $Pto = 0$ 
5:   for  $b = Orientacion$  to  $Pto < 4$  do
6:     switch  $b$ 
7:     case -4:
8:        $Pox = Pox + 1$ 
9:        $Pto = Pto + 1$ 
10:    case -3:
11:       $Poy = Poy + 1$ 
12:       $Pto = Pto + 1$ 
13:    case -2:
14:       $Pox = Pox - 1$ 
15:       $Pto = Pto + 1$ 
16:    case -1:
17:       $Poy = Poy + 1$ 
18:       $Pto = Pto + 1$ 
19:    case 0:
20:       $Pox = Pox + 1$ 
21:       $Pto = Pto + 1$ 
22:    case 1:
23:       $Poy = Poy + 1$ 
24:       $Pto = Pto + 1$ 
25:    case 2:
26:       $Pox = Pox - 1$ 
27:       $Pto = Pto + 1$ 
28:    case 3:
29:       $Poy = Poy - 1$ 
30:       $Pto = Pto + 1$ 
31:    case 4:
32:       $Pox = Pox + 1$ 
33:       $Pto = Pto + 1$ 
34:    case 5:
35:       $Poy = Poy + 1$ 
36:       $Pto = Pto + 1$ 
37:    case 6:
38:       $Pox = Pox - 1$ 
39:       $Pto = Pto + 1$ 
40:    case 7:
41:       $Poy = Poy - 1$ 
42:       $Pto = Pto + 1$ 
43:    if  $Contorno(Pox, Poy) > 0$  then
44:      salir del ciclo For
45:       $orientacion = orientacion - 1$ 
46:       $Cv = Cv + 1$ 
47:       $P_{sx} = Pox$ 
48:       $P_{sy} = Poy$ 
49:       $Contorno(Pox, Poy) = Contorno(Pox, Poy) - 1$ 
50:    else
51:       $orientacion = orientacion + 1$ 
52:       $Pox = P_{sx}$ 
53:       $Poy = P_{sy}$ 
54:    end if
55:  end for
56: end while
57:  $Cvs_{(cuenta)} = Cv$ 
58:  $cuenta = cuenta + 1$ 
59:  $orientacion = Cv = 0$ 
60: if Fin del fichero then
61:    $Cv_{fin} = Prom(Cvs(cuenta))$ 
62: end if

```

Utilizando los resultados del algoritmo LDSI expuestos en la tabla 4.1, se procedió a evaluar los resultados que, aunque ya eran satisfactorios mediante una comparación del factor de reconocimiento del software Sherlock, se analizaron mediante los algoritmos ENM y ESM con el fin de obtener una metodología diferente de comparación.

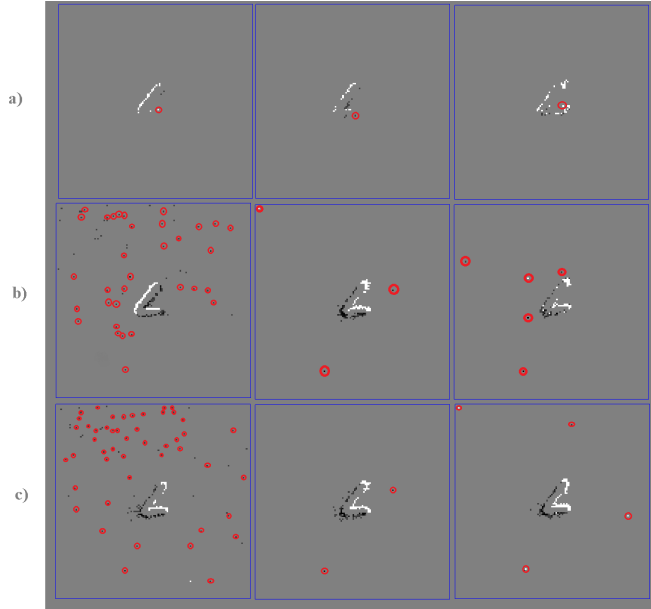


Figura 4.18: Secuencia de eventos bajo las tres parametrizaciones del algoritmo LDSI. Se señalizan los eventos que el algoritmo ENM ha detectado como ruido. a) eventos detectados como ruido bajo el filtrado con parámetros más estrictos del LDSI, b) eventos que ha detectado como ruido con parámetros medios del algoritmo LDSI y finalmente, c) con la parametrización más débil en cuanto al filtrado LDSI.

En la figura 4.18 se muestra una secuencia de tres capturas de momentos consecutivos de los datos de salida de tres parametrizaciones diferentes del LDSI. En a) se observa el resultado tras aplicar una configuración restrictiva en la generación de eventos y detección de ruido del algoritmo LDSI, en b) se representa una configuración menos restrictiva y finalmente en c) se refleja la menos sensible a la detección de ruido. Se ha marcado en un círculo en cada secuencia lo que el

Tipo de filtro	Tamaño (bytes)	ENM(eventos)	ESM(Longitud)
Secuencia Original	153576	3528	144
Filtro Débil	103176	1080	140
Filtro Medio	51696	843	133
Filtro Restrictivo	22614	51	72

Tabla 4.2: Resumen comparativo entre las diferentes parametrizaciones del algoritmo LDSI, y de las capacidades de identificación de ruido del algoritmo ENM y de la similitud del tamaño de bordes de ESM. Se hacen tres comparaciones, el tamaño del fichero que contiene los eventos, el número de eventos considerados como ruido y la longitud promedio de los bordes del objeto en escena.

algoritmo ENM ha detectado como ruido. Es posible observar que en a), al ser el resultado tan restrictivo, se han generado pocos eventos y parted del borde del objeto en la escena son identificados como ruido, aunque son relativamente pocos en comparación del total de eventos generados. Por el contrario, en c), debido a que el algoritmo LDSI eliminó menos eventos considerados como ruido, el algoritmo ENM tiene la capacidad de identificarlos como ruido. En la tabla 4.2 se resumen los resultados obtenidos.

Por parte, en la tabla 4.2 se puede observar que el tamaño de los bordes es similar cuando el filtrado es débil y medio a la secuencia original. Sin embargo, en el filtro restrictivo el tamaño de los bordes es significativamente menor, lo cual es apreciable en la figura 4.18 en donde en ocasiones desaparece parte de la silueta del objeto reduciendo su tamaño a comparación del original.

4.7 Resumen

El algoritmo LDSI es un filtro bio-inspirado, basado principalmente en capas de unidades interconectadas análogas a la estructura de una neurona. El algoritmo cumple satisfactoriamente con el objetivo inicial, desempeñarse como un filtro que provee un amplio rango de

resultados de filtrado. Ha sido demostrado que reducir la cantidad de datos y el ruido en diferentes niveles no solo depende de los niveles de los parámetros configurados en el algoritmo y la interacción entre ellos sino también de las condiciones del movimiento en la escena, siempre considerando que este movimiento es relativo entre la cámara y los objetos en la escena. Movimientos rápidos producen una gran cantidad de eventos, pero reducen la definición de los contornos de los objetos. Por el contrario, movimientos lentos producen una menor cantidad de eventos, y dependiendo de la velocidad del movimiento también la definición de los contornos de los objetos puede ser reducida. Por esta razón, la importancia de poder configurar los parámetros del algoritmo, adaptando la configuración a las necesidades y condiciones de la aplicación.

Los eventos de salida del algoritmo LDSI pueden llegar a ser hasta del 30 % de la captura original, e incluso más si la aplicación así lo requiere y no interesa la calidad en la definición de los bordes. Usualmente, las cámaras basadas en eventos producen mucha menos información que una cámara convencional por la propia tecnología, sin embargo, con el algoritmo LDSI aún se puede reducir más la cantidad de datos que contiene la información capturada. Esta característica convierte a LDSI en un algoritmo potente cuando en el mismo canal de comunicación se requieren enviar datos adicionales independientes de la cámara, como lo es una red Ethernet, ya que se genera una baja tasa de transferencia de datos, evitando la congestión de red y la disponibilidad para que otros equipos intercambien información. El algoritmo LDSI ha sido diseñado para tratar, analizar y procesar eventos, por lo que su aplicación no es solo en el ámbito de la visión artificial, podría ser aplicado a cualquier campo o sensor donde se requiera una reducción de datos/eventos manteniendo la información relevante.

El desempeño del filtro LDSI fue sustentado en primera instancia con metodologías utilizadas para datos producidos con cámaras convencionales basadas en fotogramas, mediante el software Sherlock, ob-

teniendo resultados satisfactorios y coherentes con la tecnología. Los resultados también fueron coherentes con los algoritmos diseñados para medir el desempeño de filtros como el LDSI. Los algoritmos aquí presentados, ENM y ESM, corroboraron que los resultados y los niveles de filtrado eran los esperados según las distintas parametrizaciones. Se comprobó que los algoritmos para la evaluación del desempeño del filtrado cumplían con el objetivo, al visualizar el resultado mediante jAER. Analizando dato a dato los resultados que ofrecían los algoritmos ENM y ESM podía corroborarse el rango de similitud y la detección y eliminación de ruido entre un conjunto de datos aplicando o sin aplicar el filtro LDSI.

La reducción de datos que se logra con el algoritmo LDSI manteniendo la información es una característica importante en aplicaciones en donde se requiere velocidad de procesamiento y transmisión. La transmisión juega un papel clave en el desempeño del sistema donde se utilizarán los datos resultantes de un sistema basado en eventos. Por esta razón, en el siguiente capítulo se hace un análisis de las principales posibilidades de transmisión de los datos basados en eventos en una red de comunicaciones industriales.

Capítulo 5

Análisis de Protocolos de Comunicación Basados en Ethernet para Transmisión de Eventos

Uno de los objetivos de esta tesis es el de incorporar los sensores basados en eventos dentro del entorno industrial para aprovechar de esta manera sus características en la optimización de los procesos de fabricación. Para conseguirlo, resulta necesario incorporar los datos de los sensores de eventos dentro de las redes de comunicación industriales habitualmente utilizadas. Los entornos industriales suelen incorporar equipos de diferentes fabricantes que se intercambian información a través de protocolos estándar. Existen numerosos estándares y en ocasiones no es sencillo elegir el más adecuado según la aplicación. En este capítulo se profundiza en algunas de las opciones posibles para transmitir eventos, que servirá de enlace entre los dispositivos neuromórficos y el proceso industrial.

5.1 Panorama en las comunicaciones industriales

A finales del siglo pasado la diversidad entre los diferentes protocolos de comunicación industrial no solo radicaba en la manera de codificar la información sino también la capa física por la que se transmitía, ejemplos son los estándares RS-232, RS-422, RS-485, CAN entre muchas otras variantes. No obstante, desde principios de este siglo la tendencia es utilizar Ethernet como capa física de transferencia de datos entre diferentes equipos en el sector industrial.

Ethernet ha tomado la delantera en las comunicaciones industriales, ofreciendo excelentes capacidades de compatibilidad, rendimiento y robustez que han demostrado proporcionar un marco sólido para el intercambio de información en todos los niveles industriales, desde la gerencia hasta el sensor en planta. En años recientes, con el advenimiento del Internet Industrial de las Cosas (IIoT) vinculado con el concepto Industria 4.0, la comunicación Ethernet juega un papel clave en el despliegue de máquinas eficientes, recolección de datos y estrategias de control en las fábricas de próxima generación. Sin embargo, los diferentes protocolos de comunicación que utilizan Ethernet como una conexión física común (principalmente, las capas 1 y 2 según el modelo OSI) usualmente requieren de adaptaciones según el volumen y tasa de transmisión de datos requerida. El modelo TCP/IP, derivado del modelo OSI, es ampliamente difundido y puede utilizarse para la gestión y supervisión de datos en niveles industriales, pero no es siempre la mejor opción para utilizar de manera eficiente. En la planta industrial, los controladores, sensores y actuadores deben actualizar información en muy poco tiempo, y aún más importante, en un tiempo máximo garantizado para que los procesos industriales rápidos puedan ejecutarse correctamente, concepto conocido como determinismo.

Existen múltiples propuestas de protocolos basados en Ethernet en el sector de la automatización. Todas afirman ser las más rápidas, más robustas y calificadas por la industria. Algunos ejemplos de los proto-

colos más populares y más implementados a nivel mundial basados en Ethernet son: Ethernet/IP [81], Profinet [82], EtherCAT [83], Modbus [47] y POWERLINK [84]. Todos estos protocolos se basan en Ethernet, pero con matices en la manera en que se procesan y codifican los datos en diferentes capas del modelo OSI.

Debido a que el abanico de posibilidades es bastante amplio, puede ser difícil seleccionar el protocolo adecuado si las prestaciones requeridas de comunicación no vienen acotadas por la aplicación final. Usualmente es posible encontrar comparaciones de los diferentes protocolos donde cada fabricante o principal exponente describe ventajas sobre los competidores en función de sus propios resultados experimentales [42], [85], [48], [86], y [87], claro está, estos resultados en la mayoría de las ocasiones carecen de objetividad. Por ello, es necesario realizar análisis independientes donde se analizan las redes industriales.

En este trabajo, se describen diferentes pruebas experimentales para comparar los protocolos Ethernet más consolidados en una de las ramas de la automatización más estrictas en cuanto a la velocidad de transmisión de datos: el control de movimiento. En este tipo de aplicaciones se suelen utilizar ciclos de actualización de información por debajo de los 10 milisegundos, tiempo que la mayoría de los protocolos basados en Ethernet podría cubrir, pero no de manera determinista. Los protocolos elegidos para comprobar fueron Ethernet/IP y POWERLINK. Estos protocolos son concebidos y probados para altas prestaciones de intercambio de información en sistemas de control de movimiento, y a diferencia de otros protocolos populares con también altas prestaciones en este sector como EtherCAT y Profinet IRT no requieren adecuaciones en el hardware para este propósito.

La idea principal de este estudio es establecer los límites bajo los cuales un protocolo puede cumplir con sus especificaciones de transferencia, tanto en cantidad y velocidad, bajo diversas configuraciones como a nivel software y hardware. Esto puede ser útil para establecer algunos

criterios de uso, es decir, proporcionar directrices que pueden ayudar en la decisión de utilizar un determinado protocolo cuando se sabe que la red podría estar expuesta a cualquiera de las situaciones que aquí se han probado. Antes de dar pie a la exposición de las pruebas y los resultados, se presenta en la siguiente sección una breve descripción de Ethernet/IP y POWERLINK.

5.2 Ethernet/IP y POWERLINK

Ethernet/IP y POWERLINK son dos protocolos para la comunicación entre múltiples dispositivos, máquinas y plantas industriales. Están diseñados con el objetivo de ser implementados desde el nivel de máquina hasta el nivel del proceso, con la posibilidad de involucrar las comunicaciones de la gestión de la planta. No obstante, en la práctica tienen una mayor presencia en los niveles más bajos de lo que se conoce como pirámide de la automatización [88], [89] y [90], interconectando controladores con actuadores, sensores y otros dispositivos de campo. A nivel de máquina, se requiere una respuesta de alta velocidad, mientras que, a nivel de proceso, se requiere la eficiencia en la transmisión de gran cantidad de datos; algunos ejemplos son el envío de una posición de punto de ajuste en un servomotor, el informe del estado de una máquina o sistema de automatización completo a un escritorio de supervisión central.

Ethernet/IP (Ethernet Industrial Protocol) utiliza el modelo productor/consumidor siendo *peer to peer*, es decir, cualquier equipo puede ser productor y consumidor de datos. Ethernet/IP implementa el Protocolo Industrial Común (CIP) [46] en sus capas superiores. CIP abarca un conjunto completo de mensajes y servicios para una variedad de aplicaciones de automatización de procesos y fabricación, que incluyen control, seguridad, sincronización, movimiento, configuración e información, por lo que su rango de implementación es amplio.

Por otra parte, POWERLINK utiliza un modelo maestro/esclavo modificado donde los esclavos también pueden compartir información entre ellos. Al maestro se le conoce como nodo de gestión o administrador (MN) y al esclavo como nodo controlado (CN). POWERLINK es adecuado para estructuras descentralizadas de máquina y planta para proporcionar a los usuarios mayor flexibilidad para adaptaciones y extensiones, que como se ha dicho antes, está adherido al estándar Ethernet IEEE 802.3, ofreciendo dos características clave para su uso en entornos descentralizados: *crosstraffic* y una elección libre de topología de red. Con la capacidad de *crosstraffic* se permite a cualquier dispositivo compartir información con cualquier otro en la red sin la necesidad de pasar por un nodo administrador o gestor, aunque claro, el momento de hacerlo debe estar regido por el nodo que gestiona la red. La capa de aplicación de protocolo se define como una portadora de todos los mecanismos de CANopen [91]. POWERLINK ha sido adoptado por el *Institute of Electrical and Electronics Engineers* como el estándar de comunicación industrial IEEE 61158 [92], único protocolo de comunicación industrial basado en Ethernet en obtener hasta el día de hoy dicha calificación.

5.3 Pruebas comparativas entre Ethernet/IP y POWERLINK

Por lo general, es fácil encontrar las especificaciones de rendimiento de cada protocolo industrial. Sin embargo, las pruebas que se pueden encontrar de rendimiento son realizadas por la misma organización que promueve el protocolo, por lo que las conclusiones pueden resultar sesgadas, sin resultados independientes. Por esta razón, se exponen cinco pruebas que intentan simular las diferentes circunstancias que se pueden encontrar en los entornos industriales y de las cuales se pueden ver afectadas las redes de comunicación en una planta industrial. Las pruebas diseñadas fueron las siguientes:

1. Análisis del tiempo de respuesta mediante la transferencia de la cantidad mínima de datos.
2. Análisis del tiempo de respuesta mediante la transferencia de la cantidad máxima de datos.
3. Análisis de la compatibilidad de la red en conjunto con una red corporativa convencional.
4. Análisis de resistencia contra ciberataques.
5. Análisis del tiempo de recuperación después de un fallo del bus.

Estos análisis proporcionarán similitudes y diferencias entre Ethernet/IP y POWERLINK, para facilitar la tarea al ingeniero de automatización a la hora de tomar una decisión acertada al implementar una red de comunicación en una máquina o planta industrial. En lo que concierne a esta tesis, se consideran importantes en cuanto al tiempo mínimo esperado en transportar un evento, la cantidad máxima de eventos a codificar en un solo paquete de datos y la compatibilidad de intercambiar datos de dispositivos basados en eventos en redes que convergen en mismo entorno industrial. También se prueban las capacidades a la resistencia a fallos de transmisión ajenos a la red, como puede ser un ataque de suplantación de identidad del dispositivo o un fallo en algún dispositivo de la red que entorpezca el intercambio entre otros dispositivos.

El equipo utilizado en las pruebas incorpora la última tecnología que es posible encontrar en el mercado, concretamente para entornos industriales que emplean protocolos de comunicación, asegurando de esta manera tener una base fiable y un comportamiento reproducible en la industria.

Para las pruebas del protocolo Ethernet/IP, se utilizaron dispositivos del fabricante japonés *OMRON Corporation*. Específicamente, los modelos utilizados fueron un controlador NJ501-1500 [93] y un NJ101-

9020 [94], ambos programados y configurados con SYSMAC STUDIO V1.17 [95]. Las pruebas para POWERLINK se realizaron en dos controladores X20CP1382 [96] del fabricante austriaco *Bernecker and Rainer Industrie-Elektronik*, B&R. En este caso, el software utilizado para la programación y la configuración AUTOMATION STUDIO 4.2.2.134 [97]. Para poder realizar mediciones relevantes para cada prueba, se utilizaron un conmutador gestionado TP-LINK modelo TL-SG3210 [98] y un ordenador portátil Fujitsu AH544 con un procesador i7-4702MQ y 4 GB de RAM. La configuración del hardware y la estructura de la red para ambos protocolos se muestran en la figura 5.1.

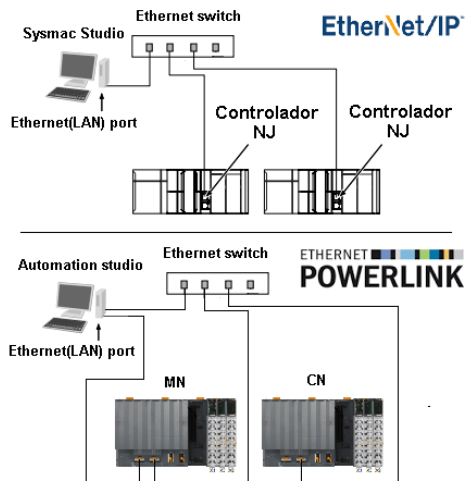


Figura 5.1: Arriba: configuración de hardware para probar Ethernet/IP. Es posible programar ambos dispositivos; esta configuración también es válida para probar el rendimiento del protocolo a través del mismo puerto. Abajo: configuración del hardware para probar POWERLINK. A diferencia de Ethernet/IP, es necesario usar un puerto para comunicarse con el software Automation Studio para programar los dispositivos y otro puerto para la prueba de protocolo. MN representa el nodo de administración y CN el nodo controlado.

En la figura 5.1 se observa que el intercambio de datos se hace a través de un switch gestionable, que por medio de la característica de *port mirroring*, se puede acceder a los datos que no son broadcast entre los dos controladores. De lo contrario, solo podríamos acceder desde el ordenador a datos que sean multicast o unicast.

5.4 Resultados

Los resultados de las cinco pruebas propuestas se muestran en las siguientes subsecciones.

5.4.1 *Análisis del tiempo de respuesta mediante la transferencia de la cantidad mínima de datos*

Considerando que Ethernet/IP es una comunicación *peer to peer*, en donde cualquier dispositivo puede generar y requerir información a través de la red, se crearon dos variables en cada dispositivo, una variable se creó como datos producidos y la otra como datos consumidos, ambas de tipo booleano. Sin embargo, aunque el bit es la mínima expresión de codificación de información, no es posible transferir solo un bit en Ethernet/IP y debe estar empaquetado en al menos un registro de 16 bits. El ciclo de ejecución o ciclo de scan de ambos dispositivos es de un milisegundo, y el intervalo de paquetes solicitado (RPI) es el mínimo permitido, de un milisegundo también.

El tiempo analizado es el transcurrido desde que los datos son producidos hasta que se genera una reacción en la variable consumida por el dispositivo. El diagrama de intercambio de datos se muestra en la figura 5.2 donde el objetivo es medir la diferencia de tiempo $T1 - T0$. Es importante considerar que entre los tiempos marcados $T0$ y $T1$ hay más datos producidos y consumidos por los dispositivos pero que pertenecen a las respuestas y peticiones de datos anteriores a $T0$ y

que durante el tiempo expuesto el controlador también se encuentra procesando.

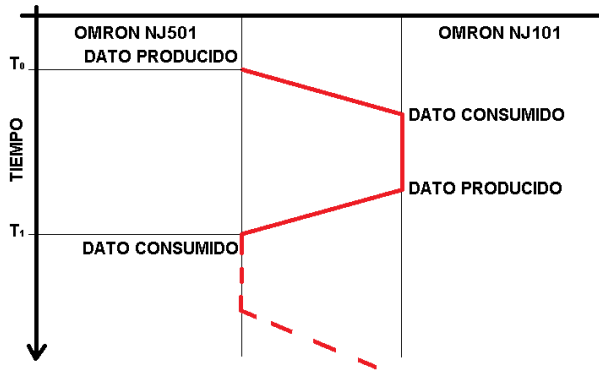


Figura 5.2: Ethernet/IP. Diagrama de intercambio de datos. Los datos son enviados por el NJ501 como productor, al NJ101 como consumidor; luego, los datos son utilizados por el NJ101 y producen una nueva información que se consumirá por el NJ501. Este tiempo de reacción es el tiempo desde que los datos se producen en un dispositivo hasta que se genera el valor de procesamiento de la contraparte y regresa al productor original.

Los resultados de los tiempos medidos T_1 - T_0 para 100 pruebas se muestran en la figura 5.3. Como se aprecia en la figura 5.3, el tiempo medido es muy estable para todos los ensayos con un valor promedio para 100 muestras de 2.895 milisegundos y una desviación estándar de 0.069 milisegundos.

Respecto a POWERLINK, al tener su base de ejecución en el modelo maestro/esclavo, el maestro solicita información al esclavo, pero en POWERLINK el esclavo puede compartir información con otros esclavos. Para esta prueba, se crearon también dos variables booleanas en cada uno de los dispositivos. Una variable de entrada y una variable de salida con un canal de comunicación reservado para cada uno. Al igual que en Ethernet/IP, no es posible enviar un booleano a través de una comunicación Ethernet, pero a diferencia de Ethernet/IP, los datos booleanos se pueden empaquetar en solo 8 bits. Para todos los

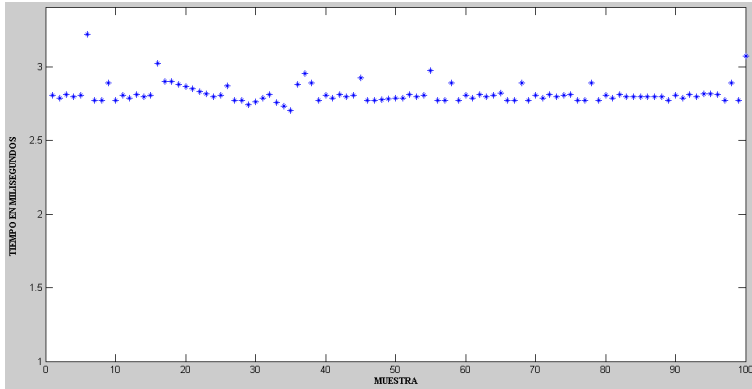


Figura 5.3: Ethernet/IP. Tiempo de reacción medido desde que el productor coloca los datos en la red hasta que el valor de estos datos toma efecto en el dispositivo consumidor y este, a su vez, devuelve los datos como un productor. En esta prueba, el tipo de datos mínimo disponible en los dispositivos se configura para ser transferido, es decir, un dato booleano (un bit).

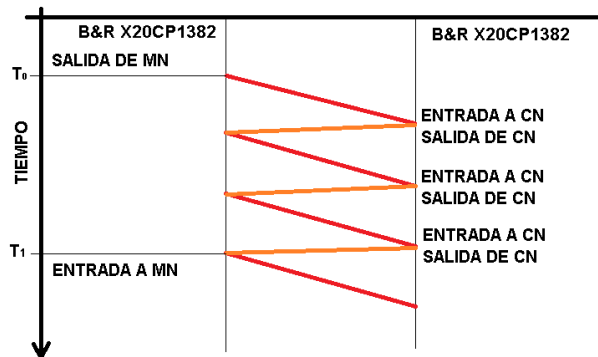


Figura 5.4: POWERLINK. Tiempo desde que el maestro pone datos booleanos agregados a un esclavo específico hasta que el esclavo procesa el valor de estos datos, que, a su vez, devuelve los datos. Este tiempo medido es el valor $T1-T0$. Al igual que con Ethernet/IP, el nodo Administrador y el nodo controlado envían y reciben datos de la red entre los dos instantes marcados. El test tiene en cuenta el tiempo transcurrido hasta que el valor enviado por el MN afecte al CN y el resultado regrese al MN.

ciclos de escaneo y actualización de entradas y salidas de los dispositivos, se fijaron al mínimo posible, un milisegundo. El diagrama de intercambio de datos se muestra en la figura 5.4.

En este caso, los resultados de los tiempos medidos se muestran en la figura 5.5. Se pueden observar tiempos muy similares a los de Ethernet/IP. El valor promedio de 100 muestras fue 2.986 milisegundos y una desviación estándar de 0.061 milisegundos.

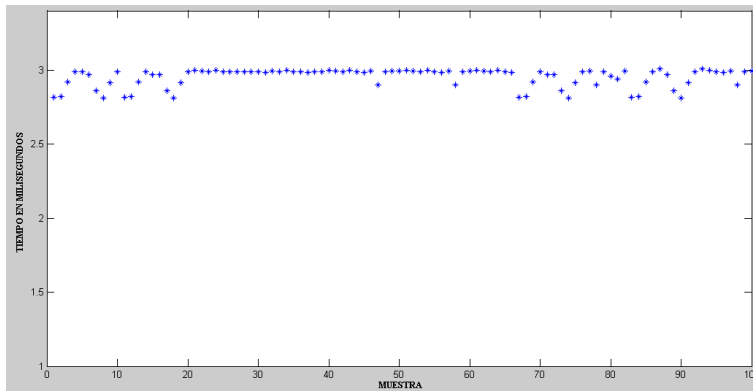


Figura 5.5: POWERLINK. Tiempo desde que el nodo de gestión (MN) coloca los datos en la red para un nodo controlado (CN) específico hasta que el nodo controlado procesa el valor de estos datos y este a su vez devuelve los datos al nodo de gestión. El nodo de gestión no solo envía datos al nodo controlado, sino que también envía un datagrama de inicio de ciclo, los datos de solicitud de información al nodo controlado y el datagrama inicio de datos asíncronos.

5.4.2 Análisis del tiempo de respuesta mediante la transferencia de la cantidad máxima de datos

Para Ethernet/IP, se creó una matriz de 300 elementos de tipo de dato palabra (16 bits), que es equivalente al máximo de 600 bytes permitidos como datos de producción en cada dispositivo. La metodología de transferencia de datos fue la misma que en la prueba anterior. Los resultados de los tiempos medidos se muestran en la figura 5.6.

En esta figura se observa que el tiempo resultante es muy similar al de la prueba anterior. Sin embargo, es posible notar mayor dispersión en los resultados. El valor promedio para 100 muestras fue 2.979 milisegundos y una desviación estándar de 0.188 milisegundos.

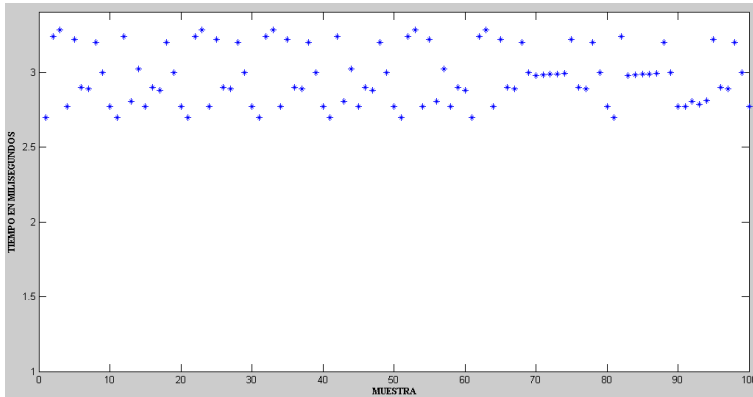


Figura 5.6: Ethernet/IP. Tiempo de reacción del bus cuando se configura la máxima transferencia de datos permitida entre ambos dispositivos (300 elementos), 600 bytes.

Para la prueba de POWERLINK, se creó una matriz de datos 'USINT' equivalente a 70 bytes, que es la máxima transmisión síncrona permitida. La matriz de datos llega al nodo controlado que devuelve los datos al nodo de administración. Por esta razón, es necesario crear una matriz de entrada y salida para cada dispositivo. La metodología de intercambio de datos fue la misma que en la prueba anterior.

Los tiempos medidos se muestran en la figura 5.7. El tiempo promedio también es muy similar al análisis anterior, en cambio, hay más dispersión alrededor del valor promedio. El valor promedio de 100 muestras fue 2.99ms y una desviación estándar de 0.07ms. Esta desviación es significativamente más baja que el resultado para Ethernet/IP lo que se traduce en un tiempo de reacción más estable y predecible en cada ciclo.

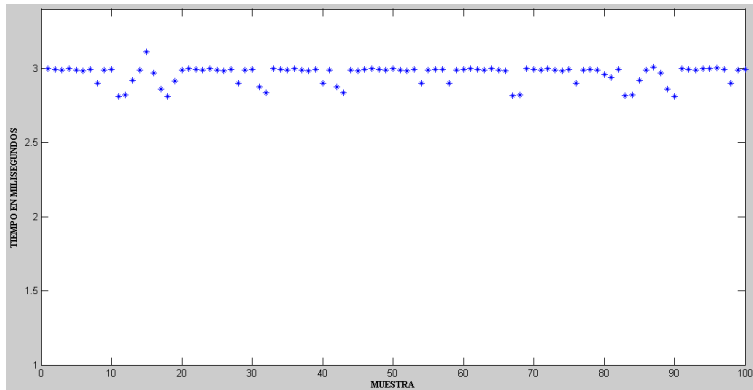


Figura 5.7: POWERLINK. Rendimiento para POWERLINK en el caso de la máxima transferencia de datos permitida entre el nodo de gestión y el nodo controlado.

5.4.3 Análisis de la compatibilidad de la red industrial convergente con una red corporativa convencional

La prueba se realizó al introducir en la misma red una conexión a Internet mediante la descarga de vídeo desde un ordenador en la misma red. El resultado en ambos protocolos fue el mismo que el obtenido anteriormente. Esto significa que, como pertenecen a subredes diferentes, el switch gestionable puede separar la información y, por ello, los datos industriales no se ven afectados, lo que prueba la solidez de las comunicaciones industriales cuando la misma línea de comunicación física es utilizada para transferir otros datos. Aquí cobra especial importancia el switch gestionable que permite separar el tráfico de datos y proporcionar un tiempo de transferencia fijo.

5.4.4 Análisis de la resistencia a los ciberataques

Para Ethernet/IP, la prueba se realizó en tres etapas principales. En la primera etapa, un dispositivo fue programado para producir datos emulando exactamente los datos como otro productor real en la red, pero usando una dirección de red diferente. El dispositivo de consumo

no resultó afectado y el bus de comunicación permaneció en estado operativo. Para la segunda etapa, también se introdujo un dispositivo en la red como productor de datos, pero esta vez con la misma dirección de red que el productor real. Una vez que el dispositivo suplantador emitió los datos a la red, el productor real emite los datos reales inmediatamente a la red en menos de 200 nanosegundos. Los datos que suplantan se introdujeron cíclicamente cada 5 milisegundos. En la tercera etapa, se realizó la misma prueba, pero reduciendo gradualmente el tiempo en el que cíclicamente se introducían los datos que suplantan a los reales, pasando de 5 milisegundos a 1 milisegundo. Cuando se introdujeron los datos suplantados, el bus ingresa en un estado de error si el tiempo cíclico de datos suplantadores era menor a 3 milisegundos.

En POWERLINK se introdujo un ataque emitiendo datos iguales a los producidos por un nodo controlado, lo que no produjo ningún cambio en los datos del nodo de gestión, pero hizo entrar en error al nodo controlado. El MN emitió un error, pero el bus aún estaba en estado de funcionamiento para los demás nodos. El bus entró en estado de error cuando los datos emitidos se enviaban cíclicamente cada milisegundo o menos.

5.4.5 Análisis de los tiempos de recuperación después de un fallo de comunicación

Una vez que cualquiera de los buses entra en un estado de error, la comunicación se detiene. El tiempo obtenido para que ambos buses vuelvan a funcionar se muestra en la figura 5.8. Ethernet/IP dio un valor promedio de 7.461 segundos y POWERLINK 5.993 segundos con una gran dispersión de la prueba, especialmente para Ethernet/IP con una desviación estándar de 5.153 segundos, mientras que con POWERLINK la desviación estándar fue de 0.507 segundos.

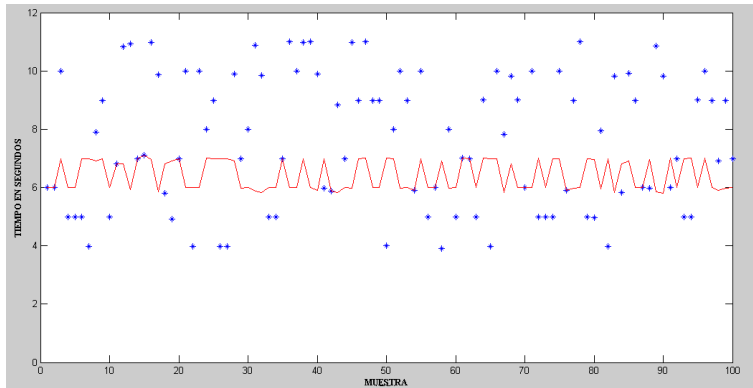


Figura 5.8: Resultados del tiempo de recuperación del bus de comunicaciones después de un error. Ethernet/IP (puntos) y POWERLINK (línea).

5.5 Resumen

Después de todas las pruebas llevadas a cabo, es posible observar algunos hechos que diferencian estos dos protocolos. Estas diferencias son factores que el ingeniero de automatización debería considerar al decidir qué protocolo implementar en un proyecto.

Ethernet/IP ofrece una gran facilidad de configuración en el intercambio de datos de forma cíclica y una gran cantidad de transferencia de datos posible por ciclo. Sin embargo, tiene una menor resistencia a los ciberataques cuando un dispositivo es suplantado, generando datos incorrectos en la red. Además, el tiempo de recuperación después de un estado de error es muy variable y dependerá de la aplicación. POWERLINK ofrece la ventaja de que puede permanecer en estado operativo cuando algunos de los nodos controlados estén siendo suplantados y omitiendo sus datos, es decir; la red se encuentra en estado operativo, sin ningún efecto no deseado cuando uno de los nodos es suplantado. Cuando el nodo administrador es suplantado, el bus entra en estado de error.

La Tabla 5.1 muestra un resumen del comportamiento de ambos protocolos de comunicación después de realizar las pruebas además de algunas características propias del protocolo. El tiempo de reacción es muy similar según los dispositivos probados pero la desviación estándar en POWERLINK fue menor, esto se refleja en una mejor estabilidad en el tiempo de reacción. Por otro lado, las tramas Ethernet/IP podrían viajar a través de varias redes hasta su destino, sin problemas, lo cual es una característica importante cuando es deseable monitorear la red desde un dispositivo remoto, característica que no es imposible con POWERLINK, pero sí limitada y difícil de poner en marcha. No obstante, POWERLINK es un protocolo completamente abierto que se puede implementar en una FPGA o microcontrolador, con códigos fuente abiertos y gratuitos [99]. Cuando se atacó la red introduciendo datos suplantados, POWERLINK tuvo un mejor rendimiento, los datos suplantados no afectaron a los dispositivos incluso cuando había dos dispositivos que emitían los datos con la misma dirección, omitiendo los datos del nodo que enviaba los mensajes fuera de su espacio de tiempo. Junto con la prueba de ciberataque, la solidez de ambos buses frente a posibles errores de los dispositivos en la red fue probada, específicamente, en el escenario en el que un dispositivo en la red sigue enviando datos cuando no está programado que lo haga. En este caso, POWERLINK permanece operativo por más tiempo y el tiempo de recuperación, una vez que el bus entra en estado de error y regresa a su estado operativo, fue más estable y predecible con solo 0.5074 segundos de desviación estándar. También es importante tener en cuenta que el proceso para la configuración de la comunicación es más rápida y simple en Ethernet/IP, aunque es un proceso dependiente del software, los pasos son menores, pues al declarar las variables se define si la variable es producida o consumida, para posteriormente asignar la asignación de relación de cada dato entre cada dispositivo, conocido como *datalink*, característica válida por ser un atributo de la variable, en cambio en POWERLINK, la variable a utilizar en

PROTOCOLO	Tiempo de Reacción	Compatibilidad Red	Defensa vs ciberataques	Robustes vs Errores	Restablecimiento	Tiempo de configuración Requerido	Código Abierto
Ethernet/IP	-	+	+	-	+	+	-
Ethernet POWERLINK	+	-	+	+	+	-	+

Tabla 5.1: Tabla resumen comparativa Ethernet/IP y Powerlink

el programa debe estar enlazada a un canal de comunicación con los dispositivos de los que se leerá o escribirá la variable.

Bajo las condiciones de configuración de hardware probadas para Ethernet/IP y POWERLINK, se concluye que ambos protocolos tienen un buen rendimiento en términos de repetibilidad y estabilidad. Ambos protocolos funcionan de manera implícita donde es necesario configurar el hardware y el espacio de memoria de los dispositivos sin necesidad de agregar líneas de código al programa del usuario. Sin embargo, Ethernet/IP es más intuitivo y se necesita menos tiempo de configuración en comparación con POWERLINK. Por otro lado, la solidez de POWERLINK frente a los ataques o errores en los dispositivos esclavos es mejor, puede permanecer en estado operativo más tiempo y en las peores condiciones de la red. Con respecto al tiempo de recuperación, POWERLINK tiene un mejor tiempo de recuperación después del ataque y la varianza del tiempo de reacción es más corta en comparación con Ethernet/IP.

Después de todas las pruebas realizadas, es posible decir que POWERLINK tiene un mejor rendimiento en entornos donde los nodos controlados pueden ser intermitentes y se requiere una menor varianza (valor fijo) en el tiempo de reacción. Ethernet/IP tiene mejores resultados en situaciones donde el acceso a la red está controlado, evitando los ciberataques y donde el tiempo de implementación se reduce debido a su rápido tiempo de configuración.

Considerando la respuesta de los análisis a cada protocolo y sus características, ambos protocolos poseen características interesantes para empaquetar datos de dispositivos basado en eventos. Ethernet/IP al

tener un modelo *peer to peer*, y permitir enviar datos a la red de manera asíncrona, parece ser una opción más afín para el empaquetamiento de eventos que POWERLINK. No obstante, POWERLINK al ser un protocolo abierto, y poder alcanzar tiempos de actualización más bajos y con la posibilidad de trabajar de manera síncrona o asíncrona brinda más posibilidades para la implementación en nuevos dispositivos, tales como los sensores basados en eventos.

Capítulo 6

Sistema de Visión Basado en Eventos en la Automatización Industrial

¿Un dispositivo generador de eventos y bio-inspirado en una línea de fabricación automatizada? Sí, y se demostrará en este capítulo la viabilidad de desarrollar nuevos dispositivos neuromórficos basados en eventos, concretamente de una cámara de visión artificial, integrándolos en una red de comunicaciones industriales. Las ventajas que esta tecnología puede ofrecer a la industria ha sido poco explorada a pesar de contar con cualidades interesantes para el sector. El sistema desarrollado para la demostración es industrial en términos del uso e integración de equipos comerciales, asimismo convergente con equipos generadores de eventos de reciente desarrollo.

6.1 Generalidades

Las cualidades de reducción de cantidad de datos generados, dimensiones, sensibilidad y consumo de energía de un sistema de visión basado en eventos pueden ser aprovechadas en un entorno industrial y este capítulo tiene el objetivo de demostrarlo. Conjuntamente se ponen a prueba nuevos algoritmos para el tratamiento de datos generados por dispositivos basados en eventos como el LDSI, incorporando en paralelo, nuevas ideas y trabajos desarrollados que se han presentado en esta tesis. Se hace uso de sistemas industriales abiertos para facilitar y motivar a más grupos de investigación la migración de dispositivos basados en eventos y sistemas neuromórficos a la industria.

Para demostrar la viabilidad de los sistemas basados en eventos y que estos interactúen con protocolos industriales modernos, se ha planteado resolver una tarea de seguimiento de un objeto que se mueve rápidamente en un espacio XY de trabajo, simulando de esta manera objetos que puedan estar pasando por una bandeja, cinta transportadora o tolva, tal como sucede en un proceso industrial. Se desarrolló la codificación y empaquetado de los eventos en una red de comunicaciones industriales para la transferencia de datos rápido, preciso y de bajo uso de ancho de banda, integrando de manera transparente el sistema bio-inspirado en el proceso industrial.

El seguimiento de objetos a altas velocidades, con movimientos erráticos o difícilmente predecibles no es una tarea fácil para los sistemas de visión convencionales basados en fotogramas, debido a que el movimiento puede ser más rápido que la velocidad de disparo de los fotogramas, desembocando en la falta de posiciones de los objetos, obteniendo una intermitencia en el cálculo de la posición y provocando incertidumbre. Una solución industrial típica al problema anteriormente planteado es el uso de cámaras de alta velocidad que ejecutan una alta velocidad de captura de fotogramas. Sin embargo, esta solución aumenta el flujo de datos en la red, haciendo necesario utilizar

redes aisladas para la transmisión de datos provenientes de una cámara hacia el controlador y por otro lado, la red del controlador con la planta. Este efecto provoca el aumento de recursos de procesamiento y almacenamiento, además del encarecimiento del controlador al precisar de mayores capacidades de comunicación. El uso de una cámara basada en eventos es una alternativa viable ya que puede proporcionar un seguimiento preciso a muy altas velocidades con menor flujo de datos que un sistema basado en fotogramas, reduciendo a la vez la carga computacional y lo que es mejor, sobre una misma red de comunicaciones, evitando el redimensionamiento del controlador.

Para este trabajo se ha desarrollado un nodo de red que configura la cámara basada en eventos; además, posee la inteligencia de configuración de la cámara, controla la generación de eventos, ejecuta el algoritmo LDSI, encapsula los eventos a transmitir por la red industrial, decodifica los parámetros que el usuario envía a través del controlador y da el soporte que requiere una comunicación industrial de este tipo, como la creación y alojamiento del fichero de descripción del dispositivo para la correcta integración con el software de programación y configuración del sistema industrial.

En el capítulo anterior se hizo un análisis entre dos protocolos de comunicaciones industriales basados en Ethernet, considerando que uno de los objetivos de este trabajo es mostrar la viabilidad de incorporar sistemas neuromórficos en entornos industriales. Para la demostración que se pretende en este trabajo se decidió optar por POWERLINK, protocolo que cuenta con las características buscadas como ser abierto, originalmente creado para sistemas de control de movimiento, con respaldo en el sector de la automatización y que al estar basado en Ethernet heredará inherentemente todas las evoluciones en lo referente a la capa física, pensando en futuras evoluciones de la tecnología.

Como se ha comentado previamente, POWERLINK es un protocolo para la comunicación entre múltiples dispositivos industriales, máqui-

nas y equipos de control. Está diseñado con el objetivo de ser implementado desde el nivel de la máquina hasta el nivel del proceso, involucrando siempre las comunicaciones de la planta industrial. A nivel de actuadores y sensores proporciona una respuesta de alta velocidad, mientras que, a nivel de proceso, facilita una eficiencia en la transmisión de gran cantidad de datos. Destaca principalmente en aplicaciones de control de movimiento con servoaccionamientos.

El nodo desarrollado está integrado en una red que incluye un nodo controlador administrador (MN) y dos nodos controlados (CN): un servoaccionamiento de dos ejes y una unidad de Entradas/Salidas distribuida basada en PLC, que en este protocolo se denomina Nodo Controlado Inteligente (iCN). La diferencia con un nodo controlado convencional es que posee más inteligencia que solo reportar estado de puntos de entrada y salida, siendo capaz de controlar por si solo un subproceso adicional. Por otra parte, el controlador gestiona la comunicación entre los nodos POWERLINK pero también incluye el cálculo en tiempo real de la cinemática inversa para un robot de dos ejes, también desarrollado exclusivamente para este trabajo. En la siguiente sección, se hace una descripción de la configuración de dispositivos para poder realizar la integración de todos los equipos.

6.2 Configuración Hardware

Para poder hacer una evaluación lo más objetiva posible al incorporar un dispositivo neuromórfico en un sistema de automatización, se plantearon dos configuraciones de sistemas de visión, uno basado en eventos y el otro basado en fotogramas y así poder comparar el desempeño de ambos. Estos sistemas convergen con la red de comunicaciones diseñada para poder controlar un mecanismo robótico. Cada sistema de visión controla de manera independiente el mecanismo diseñado y fabricado explícitamente para estas pruebas. Con ayuda de una inter-

faz gráfica es posible seleccionar qué sistema controlará la posición a la que debe ir el robot.

Con los objetivos ya fijados, se construyó el sistema industrial necesario para la demostración. Incluye un armario eléctrico industrial con dos paradas de emergencia conectado a un dispositivo de emergencia cumpliendo con las normas de seguridad industrial. Se colocaron protecciones eléctricas, fuentes de alimentación, señales de potencia y de control y todo el cableado para cumplir con los requisitos industriales. El esquema que describe los equipos y las interconexiones utilizadas se muestra en la figura 6.1.

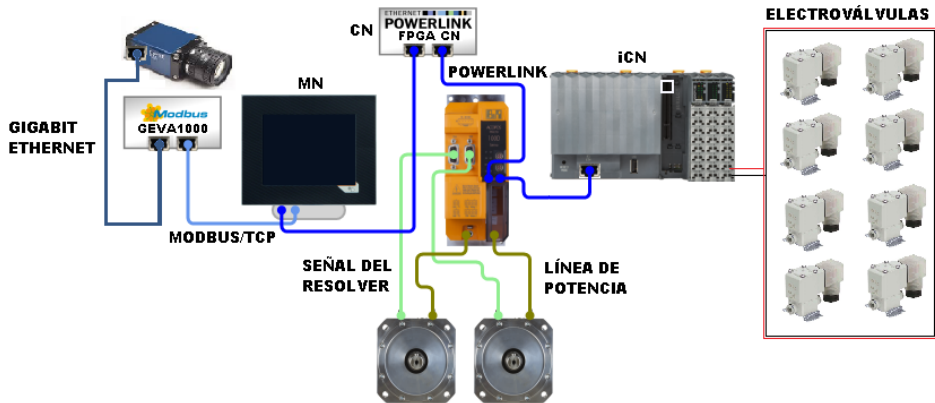


Figura 6.1: Vista general de la red de comunicaciones propuesta con los dispositivos utilizados. La cámara basada en fotogramas comunica con un PC industrial GEVA1000 por medio de GigabitEthernet. El PC industrial envía los datos mediante MODBUS TCP al nodo administrador (MN), este nodo controla por POWERLINK al servoaccionamiento (CN) que conecta los motores del eje del robot, además de recibir información del nodo FPGA (CN) de la cámara basada en eventos. También controla las salidas digitales de un nodo controlado inteligente (iCN).

La configuración hardware se observa en la figura 6.2, de la cual a continuación se detalla cada una de las partes características y funciones:

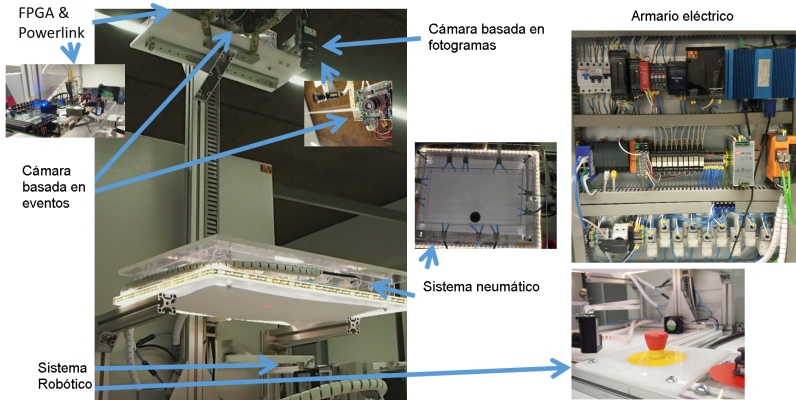


Figura 6.2: Configuración hardware del sistema para realizar las pruebas. Se remarcan las principales características del sistema. También se observa la colocación de la cámara basada en fotogramas. Se observa el contenido del armario eléctrico donde se aprecia que solo se utilizó material industrial para poder obtener los resultados más reales y objetivos posibles del comportamiento de un sistema neuromórfico en una red industrial.

- Un PLC/HMI B&R (MN), Power Panel C70, 5.7 pulgadas con una interfaz POWERLINK, una interface Ethernet 10BASE-T/100BASE-TX y dos puertos USB. En la figura 6.3 se observan las principales ventanas diseñadas en la HMI. Las funciones que desempeñará este dispositivo serán las siguientes:
 - Nodo de administración en la red POWERLINK y esclavo en la red MODBUS/TCP.
 - Encargado de recibir datos del sistema de visión artificial basado en fotogramas a través de MODBUS/TCP a través puerto Ethernet genérico.
 - Responsable de intercambiar la información con los nodos controlados por POWERLINK a través de su puerto de interfaz POWERLINK.

- Debido a que el controlador incluye un panel táctil, este sirve como interfaz para la visualización de datos y la configuración de parámetros por parte del usuario. Los parámetros que se pueden cambiar son la sensibilidad del filtro del sistema basado en fotogramas, los ajustes del algoritmo basado en eventos (LDSI), la velocidad y control del sistema neumático que mueve el objeto además de la posibilidad de poder mover el robot en forma manual o automática, esta última a su vez puede estar en referencia al sistema basado en fotogramas o en eventos.
- Calcula las ecuaciones cinemáticas necesarias para el posicionamiento del robot y envía la posición objetivo al accionamiento del motor de dos ejes para el movimiento del robot de acuerdo con la opción de la referencia del posicionamiento del robot.
- Dentro de la configuración del sistema neumático, además de activar cada electroválvula manualmente, es posible definir tendencias de trayectorias automáticas para el objeto, además de la velocidad a la que conmutaran la apertura y cierre de aire.
- La posición del objeto se puede observar en el HMI con fines informativos, aunque es importante considerar que conlleva un retraso entre la posición real y la mostrada debido a los ciclos de actualización de la visualización.
- Un PLC B&R X20CP1382 como nodo controlado inteligente (iCN) para control de entradas y salidas distribuidas. Incluye catorce entradas digitales, cuatro salidas digitales, dos entradas analógicas, dos entradas USB, un puerto RS232, un puerto bus CAN, un puerto POWERLINK y un puerto Ethernet 10/100 Base-T [96]. Tiene las siguientes características:

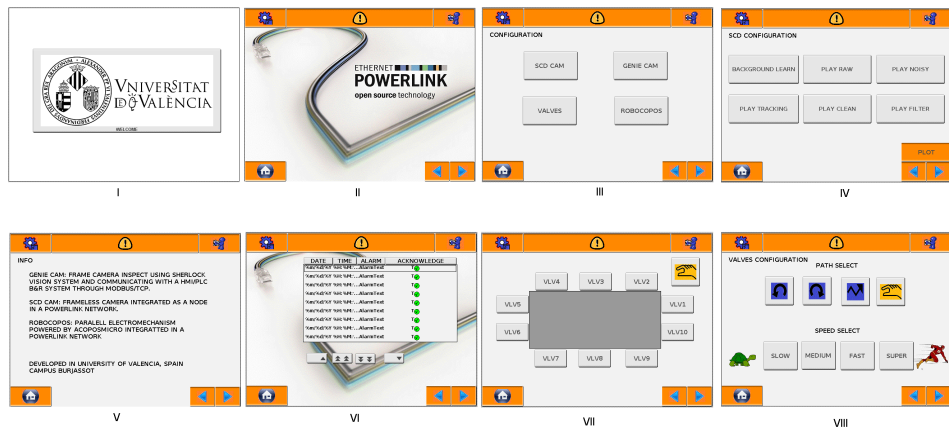


Figura 6.3: Vista general de las ventanas diseñadas en la HMI. Se tiene una ventana raíz desde la cual se puede ir a configuración de los parámetros del algoritmo LDSI, o del sistema de visión convencional, el control manual del robot y la selección de referencia para el control automático, además del control de las electroválvulas que moverán el objeto.

- Sirve como nodo controlado de entradas y salidas distribuidas en la red POWERLINK.
- Controla el sistema neumático para el soplado de aire a la bola, con diferentes trayectorias en las que se define el sentido de giro.
- Las salidas digitales están conectadas a los respectivos relés para la conmutación de las electroválvulas. Se utiliza una salida adicional para la iluminación con LED de la mesa donde se moverá el objeto.
- Joystick conectado a dos entradas analógicas, que servirán para poder posicionar manualmente el robot en una posición concreta. Se utiliza una entrada para la posición en X y otra para la posición en Y, y de esta manera cubrir el plano de trabajo del robot.

- Un módulo ANYBUS HMS M40 POWERLINK [100].
 - Sirve como pasarela de comunicación entre la FPGA y la red POWERLINK.
 - Por medio del software POWERLINK XDD FILE UPLOADER[101] y con los objetos configurados y programados en la FPGA, se puede generar el fichero de descripción del nodo. Al igual que en la mayoría de los buses de comunicación industrial, este fichero es vital para poder incorporarlo en una red de comunicaciones industriales.
- Una cámara basada en eventos de 64x64 píxeles presentada en [18]. Su función es producir los eventos captados desde la escena a la que enfoca. Esta cámara tiene la capacidad de, además de producir eventos, dar el valor del píxel en escala de grises. Puede generar un evento hasta en 2 microsegundos. La tecnología propuesta en esta cámara se ha llamado SCD *Selective Change Driven*.
- Un servoaccionamiento B&R 2-axis ACOPOS micro 100PD.022 como CN, con interface POWERLINK, con dos entradas para resolver, dos conexiones para motor, dos entradas digitales 24 VDC [102]. Tiene las siguientes funciones:
 - Control de potencia de los motores que mueven los ejes del robot.
 - Cierra el lazo de posición y velocidad de los ejes del robot.
- Dos motores síncronos B&R 8LVA13R0030D000, con velocidad nominal 3000 rpm y 0.32 Nm de par, para la transmisión de movimiento a los ejes del robot [103]. Su función es mover los ejes del robot, además de retroalimentar la posición actual al servoaccionamiento, a través de un resolver incorporado.

- Una placa de desarrollo de FPGA Zedboard [104] que contiene el *System on Chip* Zynq®-7000. Es el elemento donde mayor desarrollo se ha realizado. En la figura 6.4 se presentan los módulos programados dentro de la FPGA. Por un lado, se programó la interfaz de control y adquisición de datos de la cámara SCD basada en eventos. Por otro, se diseñaron dos alternativas de comunicación entre la FPGA con el módulo HMS M40, una puede ser por SPI y la otra por UART/serie. Para las pruebas aquí presentadas se utilizó SPI por tener una mayor velocidad de transmisión, 3Mbps, aunque la UART es más sencilla de implementar. La FPGA reporta los eventos a la red a través del módulo HMS M40, pero también la red puede solicitar datos a la FPGA, por ejemplo, estado operativo del nodo, fabricante, versión, solicitar un apagado o un reinicio o algo más complejo como el fichero de descripción de dispositivo necesario para la configuración desde el software de programación MN. En una primera instancia se hace prefiltrado con los datos adquiridos de esta cámara del fondo de la escena para descartar los píxeles espurios, los eventos correctos son filtrados por el algoritmo LDSI también implementado dentro de la FPGA. Además de lo anterior, la red proporciona los parámetros que el usuario ha configurado para el algoritmo LDSI. Esto se hace posible a través de objetos de comunicación diseñados y programados en la FPGA.

Cada objeto de comunicación posee diferentes atributos y estos a su vez tienen diferentes características de lectura, escritura o ambas. A continuación, se mencionan los objetos de comunicación mínimos requeridos para la comunicación POWERLINK entre nodo controlado y nodo administrador:

Objetos POWERLINK:

- Identificador del fabricante

- Código de producto
- Número de revisión
- Número de serie
- Nombre del fabricante
- Versión hardware
- Versión software
- Tipo de dispositivo

Objetos Ethernet:

- Dirección MAC
- Nombre Ethernet
- Revisión Ethernet

Objetos Configuración de Red:

- Dirección IP
- Número de nodo

Objetos Aplicación:

- Número de conjunto de datos
- Lenguaje
- Identificador serie de los datos
- Reset remoto

Objetos Datos de Aplicación

- Nombre de conjunto de datos de proceso
- Revisión
- Número de instancias

Los siguientes atributos forman parte de los objetos de datos de aplicación de los que se debe crear uno por cada variable a transmitir o a recibir de la red.

- Nombre de instancia
- Tipo de dato
- Número de elemento
- Descriptor de acceso
- Valor inicial
- Máximo valor
- Mínimo valor
- Número de sub-elementos
- Nombre del elemento

Los datos que se reciben de red están divididos en datos de proceso y datos de servicio. Los datos de proceso son cíclicos y están divididos en datos de entrada y de salida. Los datos de entrada contienen los parámetros del algoritmo LDSI, y los datos de proceso de salida contienen las coordenadas del último evento generado. Ambos datos requieren de un proceso de preparación para envío y recepción. Los de entrada se desempaquetan y decodifican asignando el valor correspondiente a cada parámetro del LDSI, y los de salida deben ser codificados y empaquetados den-

leído es comparado dentro del listado creado en el prefiltrado, comparando si no es un evento espurio, si no lo es pasará dicho evento al filtro LDSI.

- Un PC industrial Teledyne DALSA GEVA1000 2.4Ghz Dual Core, con dos interfaces GigE, un puerto RS232, tres puertos USB, ocho entradas digitales y ocho salidas digitales [106]. Sus funciones son las siguientes:
 - Recibe las imágenes de la cámara basada en fotogramas para el procesamiento por medio del software de inspección Sherlock [68] para el procesamiento de imágenes.
 - Se conecta al nodo administrador de POWERLINK por medio de MODBUS/TCP.
 - Los parámetros de configuración del algoritmo se pueden ajustar desde la interfaz de usuario (MN) para que el usuario pueda ajustar el algoritmo.
- Una cámara industrial convencional basada en fotogramas de la marca DALSA modelo Genie M640 (CR-GEN3-C6400), con una resolución 640x480 VGA y capacidad de hasta 64 fps [107]. Su función es obtener las imágenes de la escena en forma de fotogramas.
- Sistema de control de movimiento de un objeto (pelota) con ocho electroválvulas SMC modelo VX21 [108] controladas por el nodo controlado inteligente de POWERLINK. Con una tira de leds blancos como sistema de iluminación de fondo constante. Para poder seguir la pelota se ha diseñado y fabricado un mecanismo robótico que se explica en la siguiente sección.

6.2.1 Iluminación

Como todo sistema de visión, se requiere de un diseño de la iluminación. Para verificar la sensibilidad de ambos sistemas (eventos y fotogramas) a la iluminación se definieron tres diferentes fuentes de iluminación. Una primera será constante y estará alrededor de la mesa donde se moverá el objeto (pelota), y las otras dos fuentes estarán por encima de ambas cámaras tal como se muestra en la figura 6.5.

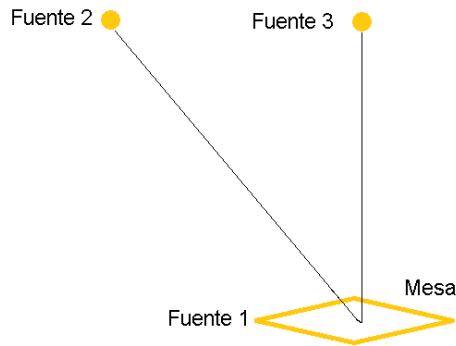


Figura 6.5: Esquema de colocación de las fuentes de iluminación, la fuente 1 es una luz de fondo constante, la fuente 2 está colocada de manera que incide a 45 grados y la fuente 3 incide a 90 grados sobre la mesa donde se desplazará el objeto.

En la figura 6.5 se representan las fuentes de iluminación del sistema de pruebas. La fuente 1 será constante y tendrá un efecto luz de fondo. Las otras dos iluminaciones se alternarán para crear diferentes combinaciones. Las combinaciones son las siguientes:

- La primera combinación que se denominará **Iluminación1** es cuando la fuente 1 y la fuente 2 están encendidas pero la fuente 3 apagada.
- La segunda combinación que se denominará **Iluminación2** es cuando la fuente 1 y la fuente 3 están encendidas pero la fuente 2 apagada.

- La tercera combinación que se denominará **Iluminación3** es cuando todas las fuentes están encendidas.

Con esto se lograrán tres diferentes intensidades de iluminación para observar la variabilidad de respuesta de los sistemas de visión ante estas configuraciones. Es necesario considerar que las fuentes de iluminación anteriores serán las controladas, pero la luz ambiente y perturbaciones como sombras y demás fuente de ruido afectarán al sistema, tal y como pasa en un entorno industrial.

En la figura 6.6 se muestra el interior del armario eléctrico resultante de los dispositivos anteriormente descritos.

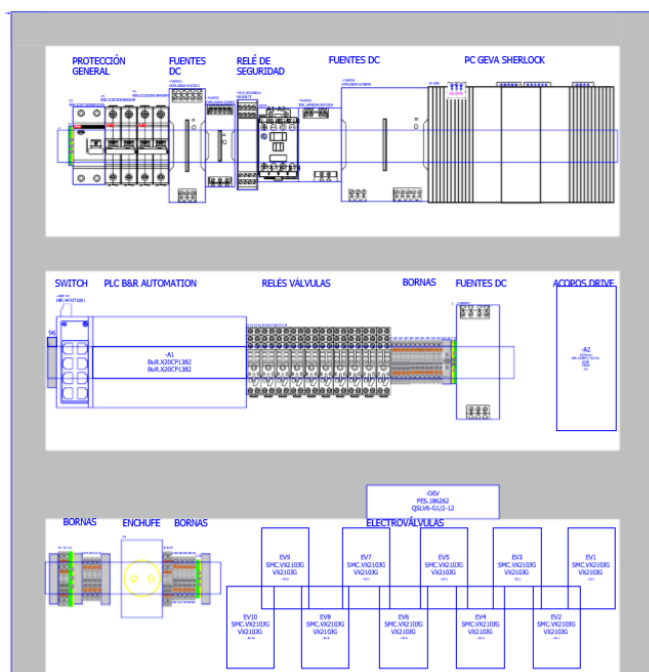


Figura 6.6: Armario eléctrico del sistema de pruebas.

6.3 Mecanismo Robótico

El mecanismo robótico propuesto para realizar las pruebas se ha diseñado con el objetivo posicionar el punto central de la herramienta comúnmente llamando TCP, en cualquier posición XY de un plano dentro del área de trabajo del mecanismo robótico. Considerando que un robot antropomorfo no sería lo bastante robusto por las velocidades que alcanza la pelota, y un robot cartesiano requeriría motores de muy altas prestaciones para alcanzar la velocidad, inviable para una aplicación de bajas dimensiones como el aquí propuesto.

El robot está formado por dos servomotores separados por una distancia D , el primer servomotor (primer eje) está ubicado en el origen $(X, Y) = (0,0)$ y el segundo motor está ubicado en $(X', Y') = (D, 0)$. Cada eje contiene una articulación, es decir, dos enlaces con longitud $L1$ y $L2$ respectivamente, que son iguales en ambos ejes. El enlace final en el primer y el segundo eje están conectados en una unión, formando el TCP. La figura 6.7 muestra el esquema y la conexión de enlaces del sistema de robot propuesto.

De acuerdo con la figura 6.7 la posición de un cierto punto (X_i, Y_i) está separada una distancia h_1 y h_2 en una línea recta desde el primer y segundo motor, con un ángulo γ y β , respectivamente. Los ángulos de rotación ξ y σ del primer y segundo motor, respectivamente, se calculan mediante la ecuación 6.1. La solución obtenida permite posicionar el TCP en (X_i, Y_i) .

Los brazos del robot se diseñaron y fabricaron explícitamente para esta aplicación, el material utilizado es aluminio y las rótulas están hechas de aleación de cobre con rodamientos planos. Es necesario señalar que se hicieron varios prototipos hasta obtener los resultados esperados, en cuanto a inercia, robustez y estabilidad. Este tipo de mecanismo ha sido muy poco explotado en la industria, a pesar de que posee

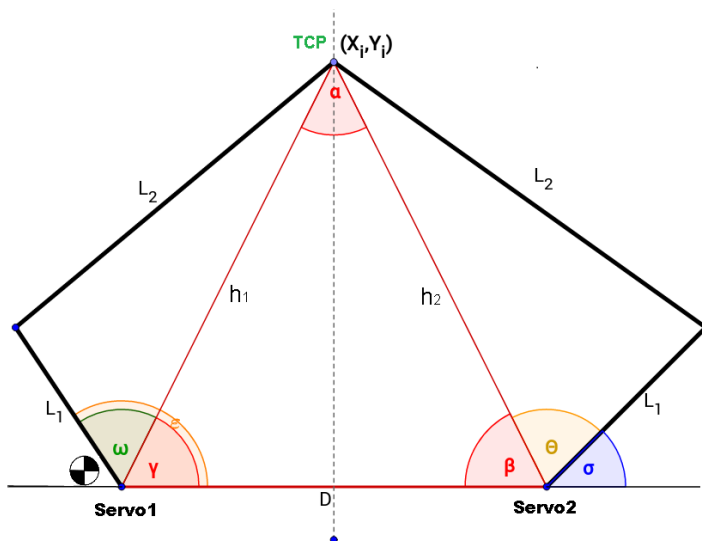


Figura 6.7: Estructura cinemática del robot propuesto. Definición de las variables para la resolución de la cinemática del mecanismo.

la ventaja de alcanzar muy altas velocidad en el TCP con motores de bajas prestaciones dinámicas en cuanto a torque y potencia disponible.

$$\begin{aligned}
 h_1^2 &= X_i^2 + Y_i^2 \\
 h_2^2 &= (D - X_i)^2 + Y_i^2 \\
 \gamma &= \arccos\left(\frac{X_i}{h_1}\right) \\
 \beta &= \arccos\left(\frac{D - X_i}{h_2}\right) \\
 \omega &= \arccos\left(\frac{h_1^2 + L_1^2 - L_2^2}{2h_1L_1}\right) \\
 \theta &= \arccos\left(\frac{h_2^2 + L_1^2 - L_2^2}{2h_1L_1}\right) \\
 \sigma &= 180 - \theta - \beta \\
 \xi &= \omega + \theta
 \end{aligned} \tag{6.1}$$

6.4 Pruebas

Se han programado ambos sistemas de visión artificial para cumplir con la tarea de seguir la pelota que se moverá empujada por el aire que expulsan las electroválvulas. Aunque la tarea parece sencilla en cuanto a definición, no implica que sea sencilla en cuanto a ejecución, debido a que la pelota además de moverse de forma aleatoria, se mueve a muy alta velocidad, lo que implica que el tiempo de refresco de la posición debe ser el mínimo posible. El mecanismo se ha diseñado para que sea estable y con poco peso en el extremo, para así favorecer la disminución de inercias causadas por las altas velocidades y cambios intempestivos de dirección de giro.

Desde la HMI, el usuario puede seleccionar el modo de operación que quiere para el robot. Los modos diseñados son: seguir una trayectoria predefinida, moverse de forma manual mediante el joystick, o seleccionar el sistema de visión que lo guiará: cámara basada en eventos (SCD) o cámara basada en fotogramas (GENIE). Para cada opción hay una pantalla para poder modificar los parámetros de filtrado.

6.4.1 Procesamiento basado en eventos

Si bien al inicio de esta sección se estableció que el objetivo de este trabajo es probar la viabilidad de los sistemas neuromórficos en el entorno industrial como parte del procesamiento de eventos, también se ha comentado la importancia de la reducción de los datos para transmitir la misma información. Es aquí donde se ha encontrado cabida a una utilidad del algoritmo LDSI planteado en el capítulo 4. Se puede decir que el objetivo de una cámara basada en eventos instalada en un entorno industrial no es emular un sistema neuromórfico, sino aprovechar algunos conceptos bio-inspirados sobre las neuronas biológicas con el fin de reducir la transmisión de datos sin pérdida de información

y mejorar el tiempo de cálculo necesario para diferentes acciones tales como reconocimiento de formas, trayectorias, posiciones etc.

A manera de resumen, se recuerda al lector que el algoritmo LDSI tiene una capa llamada 'SLAYER' que actúa como las dendritas que alimentan datos al núcleo ('DLAYER') que envían información a un gran número de terminales sinápticas ('ALAYER'). El algoritmo tiene los siguientes parámetros: ELD, ELA, ELAN, TPD, TPA, DPD, y DPA. Para una mejor descripción dirigirse al capítulo 4.

Como primer experimento, se observó que la cantidad de datos generados en la cámara de eventos sin tener ningún objeto en movimiento era importante por lo que se diseñó un sencillo procedimiento de memorización de píxeles espurios en el que el sistema procesador guardaba la posición en la que se han activado píxeles cuando el usuario le indicaba que lo que está enfocando la cámara es solo el fondo, sin ningún movimiento relativo en la escena, por lo que memorizaba la posición para posteriormente descartarla cuando entre en operación el movimiento. Uno de los motivos por el que se generan este tipo de eventos espurios ha sido investigado en [109], y obedece a efectos electrónicos propios de esta tecnología.

La cantidad máxima a memorizar y guardar sería de 500 posiciones, considerando que los $64 \times 64 = 4096$ posiciones posibles sería el total máximo admisible de ruido de la cámara. El resultado se puede observar en la figura 6.8.

Al resultado del pre-procesado fue al que se le aplicó el algoritmo LDSI, con los parámetros que desde la interfaz de la HMI seleccione el usuario.

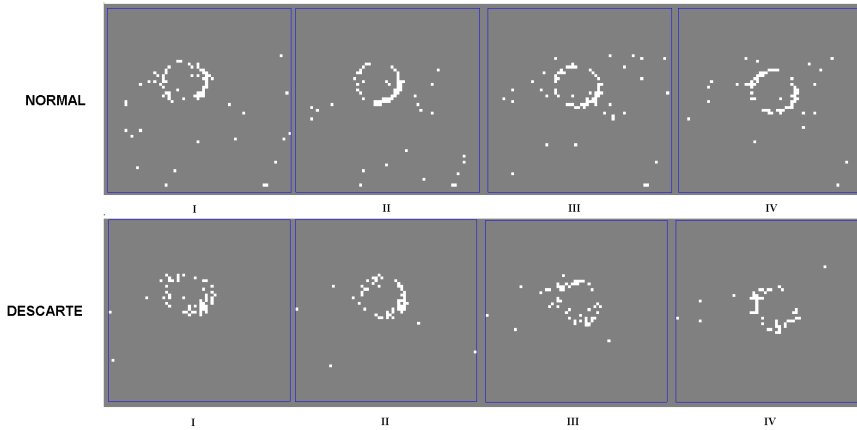


Figura 6.8: En la parte superior se muestra una secuencia NORMAL de los eventos generados sin ningún tipo de procesamiento, se observa claramente la cantidad de ruido y que algunos píxeles son repetitivos. En la parte inferior está la secuencia después de procedimiento de memorización y DESCARTE de píxeles muertos o espurios.

6.4.2 *Procesamiento basado en fotogramas*

En el PC industrial, mediante el software de visión artificial Sherlock, el procedimiento para la captura de la pelota ha sido el siguiente: se ha programado un escalado de grises, después se aplica una binarización con un nivel de umbral modificable por red, que será uno de los parámetros a poder modificar para las pruebas. Después de la binarización se utilizó la función centroide para encontrar la posición del objeto en la escena, el cual busca áreas de determinada cantidad de píxeles, previamente calibradas por el tamaño de la pelota, pero con posibilidad de modificarla desde la HMI. Finalmente, se creó una función para enviar el centroide del objeto encontrado al nodo administrador de POWERLINK por medio de MODBUS/TCP, se trató de hacer el código lo más simple y ligero de operaciones para su mayor eficiencia en el envío de la posición al controlador.

6.5 Puesta en marcha

Con la parte de visión ya definida, se dio paso a la calibración y ajustes de los servomotores del mecanismo robótico, aunque mostrar los procedimientos de sintonización y ajustes del mecanismo robótico quedan fuera de los objetivos de esta tesis se comenta brevemente lo que se hizo:

- Sintonización del controlador PI de los motores en vacío y con carga.
- Calibración de las distancias medidas por los sistemas de visión a distancias del robot.
- Creación de las rutinas necesarias en el PLC para el movimiento siguiendo las funciones estándar de programación de movimiento PLCopen [110].

La instalación de los dos sistemas de visión permite evaluar y comparar el desempeño de ambos sistemas. El hecho de que el objeto (pelota) se mueva arbitrariamente, provoca que el poder calcular el tiempo de respuesta de un sistema de visión y otro no sea sencillo, pues ambos tratan de seguir la pelota a muy alta velocidad, como es posible apreciar en los vídeos [111]. Sin embargo, cuando la pelota se detiene brevemente, la cámara basada en eventos puede perder la pelota al disminuir la cantidad de eventos a solo los píxeles espurios propios de la cámara. Este problema no se da con la cámara basada en fotogramas. Por otro lado, ante cambios repentinos de movimiento, la cámara basada en eventos reacciona casi de manera instantánea, notando que, ante grandes cambios de posición de la pelota, el sistema basado en fotogramas es menos rápido en su reacción. A pesar de tratarse de observaciones subjetivas, nos dan paso a plantearnos la siguiente hipótesis:

Los sistemas de visión basados en eventos pueden tener un comportamiento similar que el de un sistema de visión convencional basado en fotogramas cuando los objetos a capturar tienen una velocidad baja, este comportamiento mejora conforme aumente la velocidad de los objetos en movimiento.

Para poder comprobar esta hipótesis, se desarrolló un sistema experimental lo demuestre de manera más objetiva, con datos reales. Las comprobaciones y medidas deberían ser hechas de manera no intrusiva para así tener el resultado lo más real posible. La estrategia para poder conseguirlo fue la siguiente: se hizo uso de una característica diferenciadora de los PLCs de B&R, que es la función 'log' de las variables a cada paso del ciclo de scan, lo que nos permite guardar un registro de las variables de la posición del objeto a cada ciclo de scan, que en este caso era de 1.2 ms. Sabiendo que tenemos el instrumento de medida, quedaba por encontrar la forma de saber la posición real del objeto y verificar el tiempo de retraso entre ambos sistemas de visión. Lo anterior solo podría ser medido con instrumentos y equipos adicionales a los que ya se habían definido en la configuración, para ello se montó un motor que hace girar un disco blanco con un círculo negro en un extremo, así, se tendría un movimiento fijo y conocido en un plano, con forma senoidal al graficar los ejes a lo largo del tiempo. De esta manera podríamos saber cuál es la diferencia de tiempo que tarda un sistema en reportar una misma posición haciendo coincidir las diferencias de tiempo entre los máximos o mínimos de la onda senoidal. En la figura 6.9 se muestra un esquema del montaje propuesto.

Se plantearon tres conjuntos de parámetros diferentes para cada uno de los sistemas de visión, combinando con las tres diferentes iluminaciones anteriormente definidas: **Iluminación1**, **Iluminación2**, y **Iluminación3**. De esta manera se puede comparar y comprobar la diferencia de reacción a encontrar un objeto en movimiento en un plano por ambos sistemas de visión, además de verificar la sensibilidad de los mismos ante cambios de iluminación controlados y no controlados.

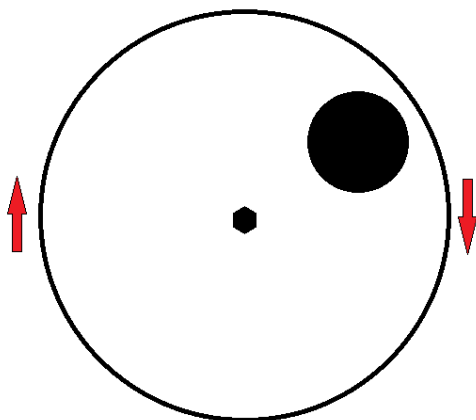


Figura 6.9: Esquema del montaje propuesto para lograr un movimiento en un plano XY conocido. Al ser un movimiento circular, el valor de los ejes en X y Y cambian gradualmente de manera repetitiva, y al graficarlos por separado a lo largo del tiempo representan una señal senoidal.

6.6 Resultados

En primera instancia se obtuvieron los eventos generados por la cámara SCD (basada en eventos) sin ningún tipo de filtro aplicado, estos datos se comparan en la figura 6.10 con los obtenidos en paralelo con la cámara GENIE (basada en fotogramas). En azul y rojo está representada la posición XY respectivamente, que calcula el sistema de fotogramas. En verde y rosa está la posición calculada por el sistema basado en eventos. Para todas las figuras que representan la posición se mantendrá este código de colores. La figura 6.10 representa los datos obtenidos por ambos sistemas bajo las tres configuraciones de iluminación diseñadas para las pruebas.

De la figura 6.10 se pueden hacer varias observaciones: una de las más claras es la cantidad de ruido y dispersión de los datos generados por la cámara SCD. Existen algunos píxeles que durante toda la prueba se mantienen activos, representados por la tendencia a crear líneas horizontales en todas las gráficas. Como se ha explicado en secciones anteriores, estos datos pueden ser fácilmente pre-procesados para poder eliminar la mayoría de los píxeles constantes y disminuir la dispersión de la señal senoidal. En la figura 6.11 se observa el resultado

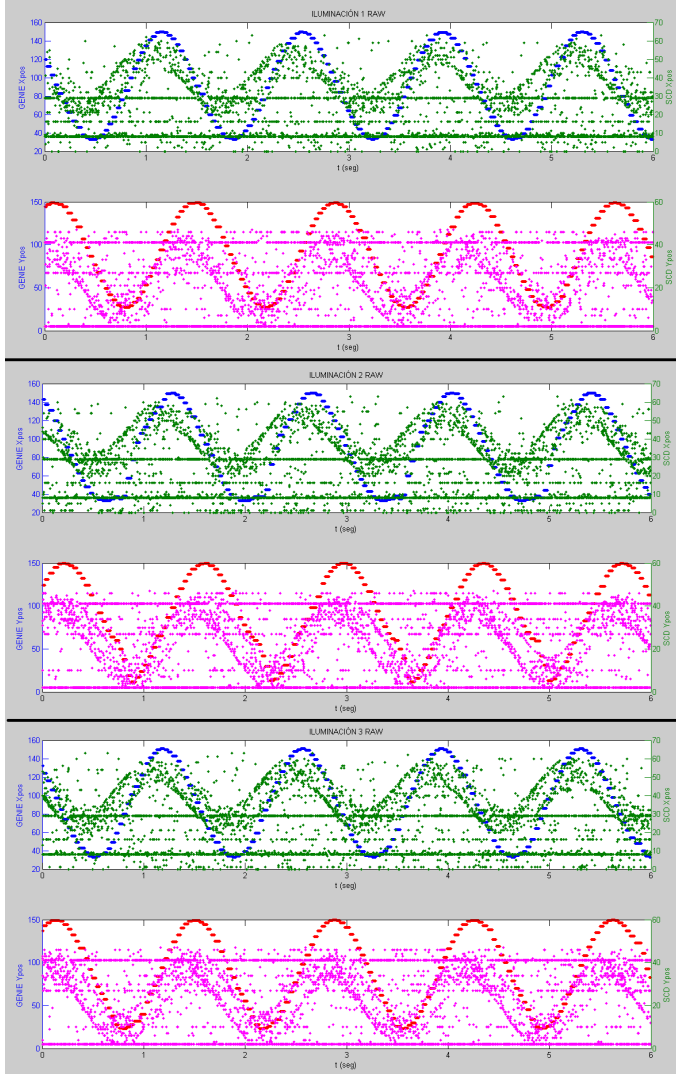


Figura 6.10: Posición en crudo reportada por ambos sistemas. En color verde y rosa los datos de la cámara basada en eventos, en color azul y rojo la posición reportada por el sistema de fotografías, ambos bajo tres tipos de iluminación diferentes. Es posible observar que a pesar de que se ve claramente el patrón de movimiento (movimiento circular) en la forma senoidal, la dispersión es grande en el sistema de eventos (SCD) además de que existe una gran cantidad de píxeles que se activan sin necesariamente representar un evento, estos llegan a ser tan constantes que representan una línea recta a lo largo del tiempo. La cámara basada en fotografías (GENIE) tiene ajustado los parámetros convenientes para cada iluminación en donde encuentra el objeto en todo momento.

Nivel	ELD	ELA	ELAN	TPD	TPA	DPD	DPA
S	1	1	1	2	2	0	0
M	3	4	4	6	6	1	1
L	3	3	3	9	9	1	1

Tabla 6.1: Parámetros Algoritmo LDSI.

tras aplicar el procedimiento de memorización y descarte de píxeles muertos o espurios.

El procedimiento de memorización de píxeles muertos o espurios tiene la ventaja de que elimina una gran cantidad de píxeles dispersos, haciendo más clara la representación del movimiento del objeto. No obstante, también es necesario tener en cuenta que eliminará los eventos que se produzcan a lo largo de todo el tiempo en las posiciones de píxeles que se han descartado, lo que según el tamaño de la matriz de píxeles puede afectar significativamente la resolución de la escena captada. En la cámara utilizada en estas pruebas, la matriz de píxeles es de 64x64, y para garantizar una resolución aceptable según el tamaño de la matriz original de píxeles se consideró un 12% del total de píxeles disponibles como tamaño máximo del arreglo de píxeles a considerar como muertos o como espurios. En la figura 6.11 la cantidad máxima de posiciones consideradas como ruido a descartar se fijó en 500.

Tras hacer el procedimiento de memorización y descarte, se aplicó el algoritmo LDSI. Como se ha definido anteriormente este algoritmo posee diferentes parámetros que afectan el resultado del número total de eventos de salida y de la sensibilidad a eliminar ruido. Para las pruebas, se hicieron tres diferentes conjuntos de parámetros para poder hacer un filtrado bajo **S**, medio **M** y alto **L**. En la tabla 6.1 se muestran los tres conjuntos de parámetros creados para las pruebas.

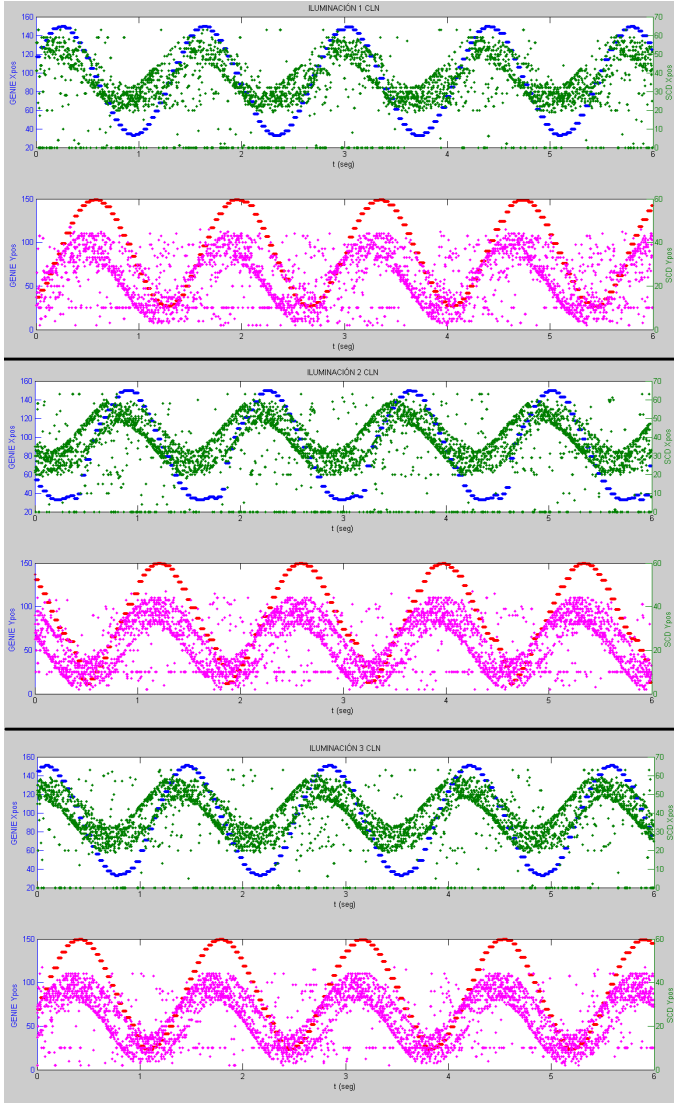


Figura 6.11: Resultados de ambas cámaras de visión. Tras aplicar el procedimiento de memorización y descarte de píxeles muertos o espurios. Es posible observar como desaparecen la mayoría de los píxeles que estaban constantemente activos, además de que se reduce notablemente la dispersión de los eventos. Sin embargo, la dispersión aún es considerable y la cantidad de datos para representar un movimiento es alta.

Consecuentemente, la cámara basada en fotogramas también necesitó de consideraciones para poder hacer las pruebas. Además del ajuste de los parámetros de umbral de cantidad de píxeles, uno de los problemas con los sistemas basados en fotogramas son los reflejos, que, aunque se presentan en objetos planos, son mucho más complicados de eliminar en objetos con bordes y formas redondeadas. En la figura 6.12 se observan imágenes captadas desde el software de visión Sherlock, en el que se muestran momentos de la captura del objeto bajo las tres iluminaciones propuestas, con los tres conjuntos de parámetros. Los parámetros son llamados de la misma manera que con el sistema de eventos: S, M y L para mayor practicidad al momento de realizar las pruebas y obtener los resultados. Para este caso, en el sistema de visión basado en fotogramas los parámetros interpretan progresivamente el valor del umbral y del tamaño del objeto a capturar. A continuación, se describen los tres conjuntos de parámetros para este sistema:

- **S** equivale a un umbral de 25 unidades de 255 y un tamaño de área máxima a capturar de 500 píxeles.
- **M** equivale a un umbral de 60 unidades de 255 y un tamaño de área máxima a capturar de 1000 píxeles.
- **L** equivale a un umbral de 100 unidades de 255 y sin límite de tamaño.

Este conjunto de parámetros fue seleccionado para poder producir resultados diferenciadores al momento de la ejecución. En la figura 6.12 se puede observar que los reflejos del fondo de la mesa donde se mueve el objeto aparecen de diferente manera para una misma iluminación, como sucede en la iluminación1 para los parámetros M y L. Por otra parte, el parámetro S con la iluminación1 fue el que de mejor manera encontró el objeto, aunque para las iluminaciones 2 y 3 el objeto desapareció completamente. Con esto se puede demostrar la alta sensibilidad a los cambios de iluminación de los sistemas basados en fotogramas, y cómo su respuesta se ve condicionada por una adecuada

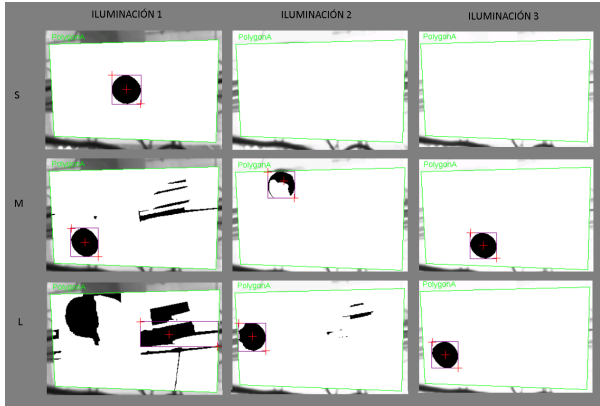


Figura 6.12: Influencia de la iluminación en la búsqueda del objeto en el sistema basado en fotogramas GENIE. En la iluminación1 con el conjunto de parámetros S, la localización es muy buena. No obstante, conforme hay más iluminación, el objeto se puede perder constantemente o aparecer solo en ocasiones, según la incidencia de las fuentes de iluminación.

parametrización. En la iluminación2 con los parámetros M, el objeto también es posible encontrarlo, no obstante, cuando la incidencia de la iluminación refleja justo a la posición de la cámara, el objeto se pierde por instantes. En la iluminación1 con los parámetros L, el objeto no solo se esconde con el fondo, sino que el sistema reconoce parte del fondo como si fuera el objeto a buscar, que para un sistema de posicionamiento puede resultar grave.

En las siguientes páginas se describe y se muestra la comparativa de los resultados obtenidos entre el sistema de visión basado en eventos SCD y el basado en fotogramas GENIE. De la figura 6.13 a 6.21 se representan en tres pares de gráficas los valores superpuestos de los datos obtenidos de ambos sistemas de visión de la coordenada X, y de la coordenada Y. Se sigue el mismo código de colores anteriormente mencionado, para el sistema basado en fotogramas el color azul y rojo, para las coordenadas XY respectivamente y los datos que provienen de la cámara basada en eventos están representados por el color verde y rosa para las coordenadas XY. Los tres pares de gráficas representan el resultado obtenido según los tres tipos de iluminación.

Es representado el resultado obtenido con la iluminación1 en la parte superior, el resultado de la iluminación2 en la parte media y en la parte inferior lo relacionado a la iluminación3.

El efecto en la incidencia de la iluminación hace mucho más sensible al sistema basado en fotogramas. En las figuras 6.13, 6.16 y 6.19 se aprecia como el sistema basado en fotogramas deja de encontrar el objeto en movimiento para las iluminaciones 2 y 3. Sin embargo con la misma parametrización con la iluminación1 funciona perfectamente obteniendo la posición del objeto de manera continua y clara.

Por otra parte, también existen combinaciones de parámetros del sistema basado en fotogramas que tienen un comportamiento irregular pero que localizan el objeto en movimiento bajo las tres iluminaciones. Este desempeño se muestra en las figuras 6.14, 6.17 y 6.20. A pesar de que la respuesta no es tan continua y nítida en la iluminación1 como en la figura 6.13, el conjunto de parámetros M del sistema basado en fotogramas GENIE, es más flexible ante cambios de iluminación.

Con el conjunto de parámetros L del sistema basado en fotogramas GENIE, se obtiene una respuesta clara y nítida para la iluminación3, una respuesta inestable pero aceptable en la iluminación2, pero un pobre desempeño en la iluminación1. Esto se puede observar en las figuras 6.15, 6.18 y 6.21.

En las figuras 6.15 y 6.21, el desempeño del sistema basado en fotogramas es diferente para la iluminación1 siendo el mismo conjunto de parámetros. Esto se atribuye a los cambios de iluminación no controlados.

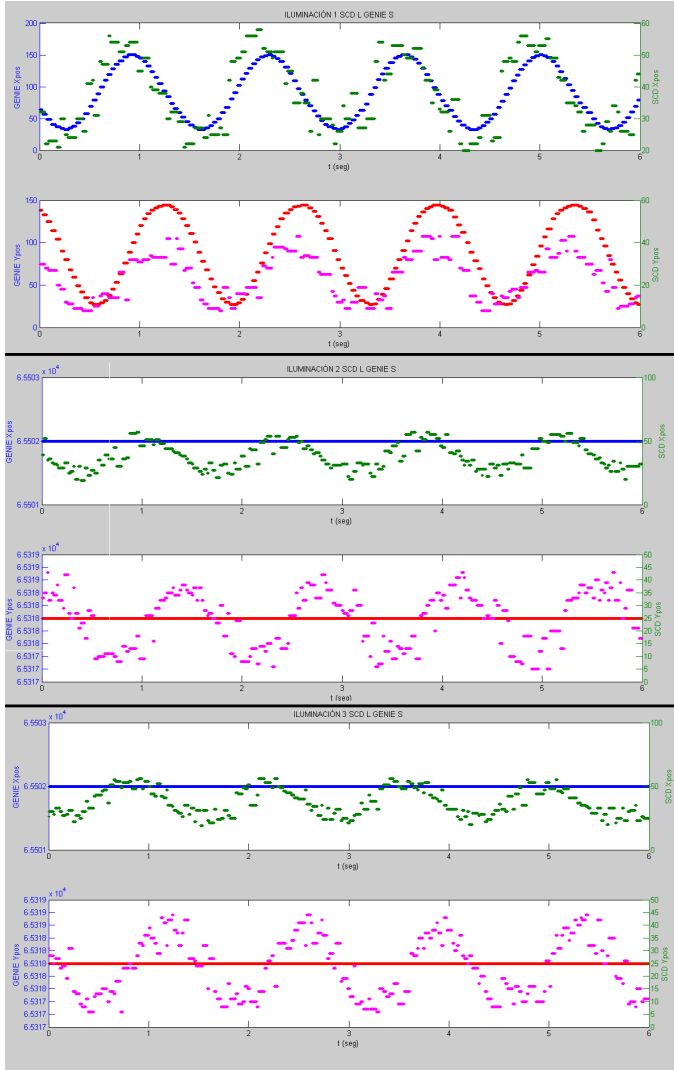


Figura 6.13: Resultado del conjunto de parámetros L del algoritmo LDSI para la cámara SCD y el conjunto de parámetros S de la cámara GENIE. Se puede observar un comportamiento limpio y continuado de la producción de datos del sistema de visión basado en fotografías para la iluminación 1. Sin embargo, para el resto de iluminaciones, los parámetros configurados en el sistema no son suficientes para poder ver el objeto en la escena. La intensidad de iluminación reduce el contraste del objeto con el fondo, haciéndolo desaparecer. La cámara basada en eventos tiene una baja producción de eventos, pero suficiente para poder observar que es sensible al movimiento y además es capaz de reportar este movimiento antes que el sistema basado en fotografías, siendo superior a 100ms la diferencia entre los dos sistemas de visión.

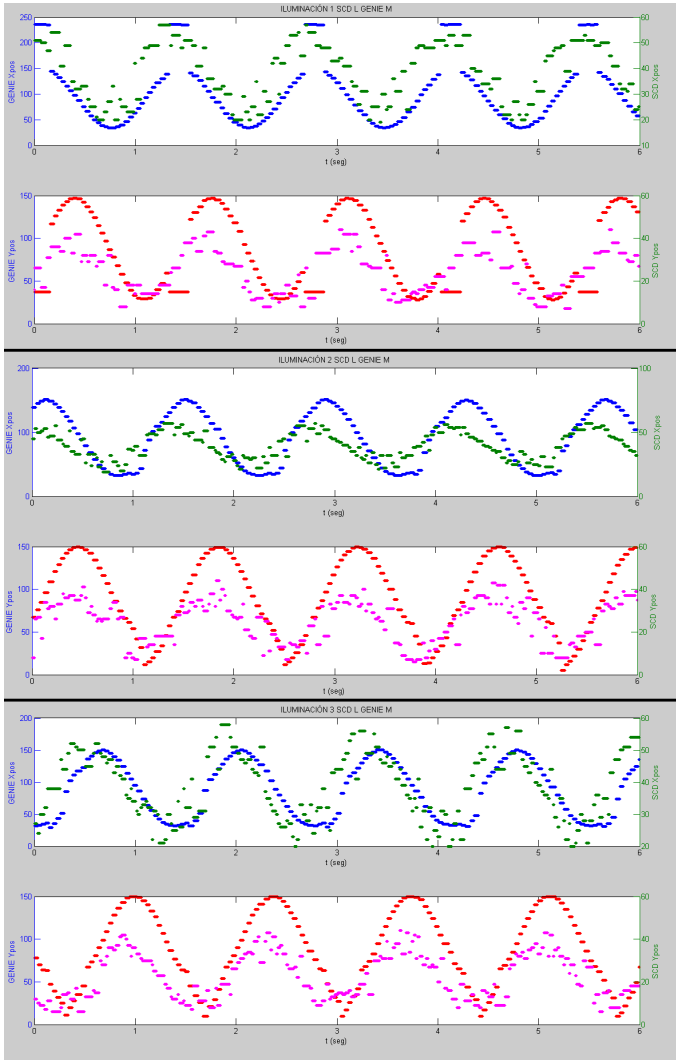


Figura 6.14: Esta figura muestra los parámetros L y M para el sistema basado en eventos y en fotogramas respectivamente. En relación con la figura anterior, el sistema de fotogramas tiene un comportamiento más estable al reconocer el objeto bajo todas las iluminaciones, pero no lo hace de una manera tan fluida como en el caso anterior, existen instantes en los que se nota una pequeña pérdida del objeto. El sistema basado en eventos nuevamente muestra que su capacidad de respuesta es mayor y se ve apenas alterado por el cambio de iluminación, teniendo una densidad ligeramente mayor en la iluminación³.

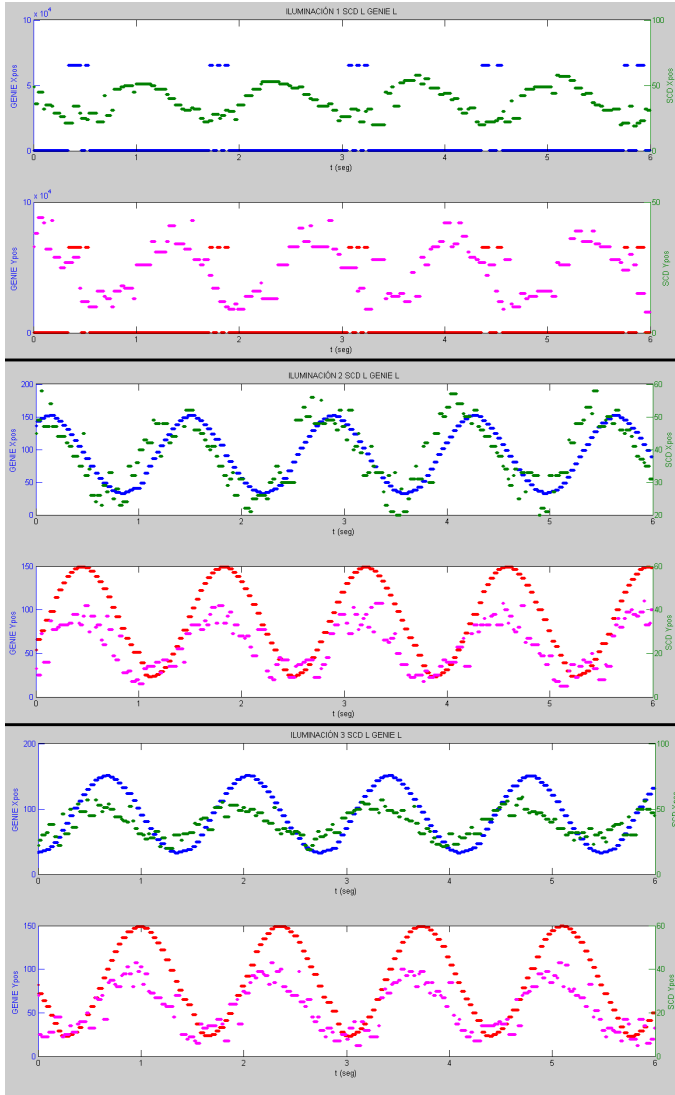


Figura 6.15: Comportamiento del sistema basado en eventos y en fotogramas para un conjunto de parámetros L. De manera similar que en la figura anterior, el conjunto de parámetros de la cámara basada en fotogramas que hace reconocer adecuadamente al objeto en escena en la iluminación 2 y 3, lo pierde en la iluminación 1, esta vez de una manera más pronunciada. En lo que al sistema basado en eventos concierne, la respuesta sigue siendo más rápida que con el basado en fotogramas, alrededor de 100ms, siendo más claro en la iluminación 3.

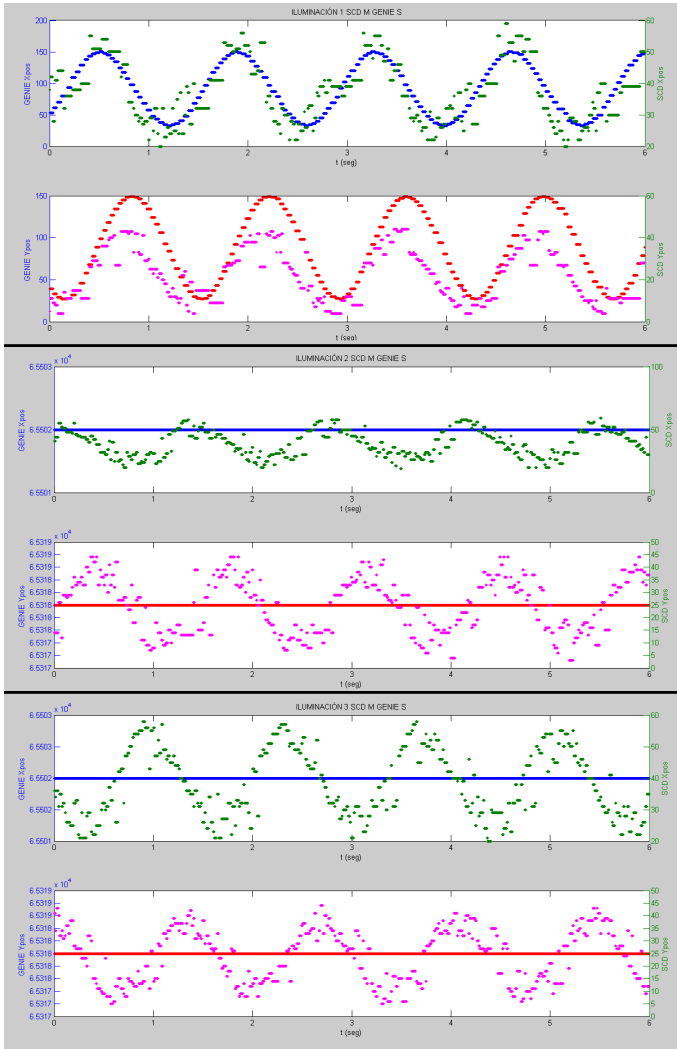


Figura 6.16: Los parámetros configurados para esta figura son M y S para el sistema basado en eventos y fotogramas, respectivamente. Los parámetros que mejor funcionan para la iluminación1 en el sistema de fotogramas hacen que pierda el objeto en las otras dos iluminaciones. La cámara basada en eventos dispersa ligeramente más que con el filtro más restrictivo, aunque no se ve afectado significativamente por el cambio de iluminación. Ligeramente se incrementa la cantidad de eventos conforme se incrementa la iluminación, al igual que con el conjunto de parámetros anteriores.

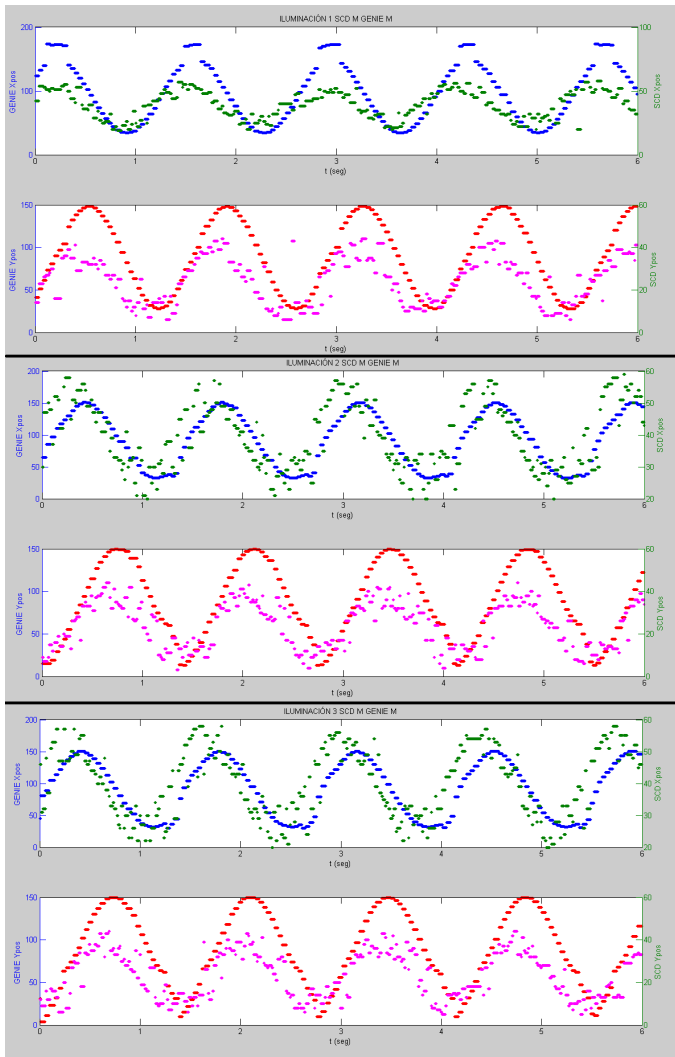


Figura 6.17: Parámetros configurados como M para ambos sistemas de visión. La cámara basada en eventos tiene mayor regularidad en encontrar el objeto que en la figura anterior. También es posible observar que la respuesta no es exactamente la misma. Este cambio es provocado por la luz ambiente que no es controlada. Los resultados son óptimos para la cámara basada en eventos, a pesar de que existe una dispersión, el valor medio es sensiblemente anticipativo al valor dado por el sistema de visión basado en fotografías.

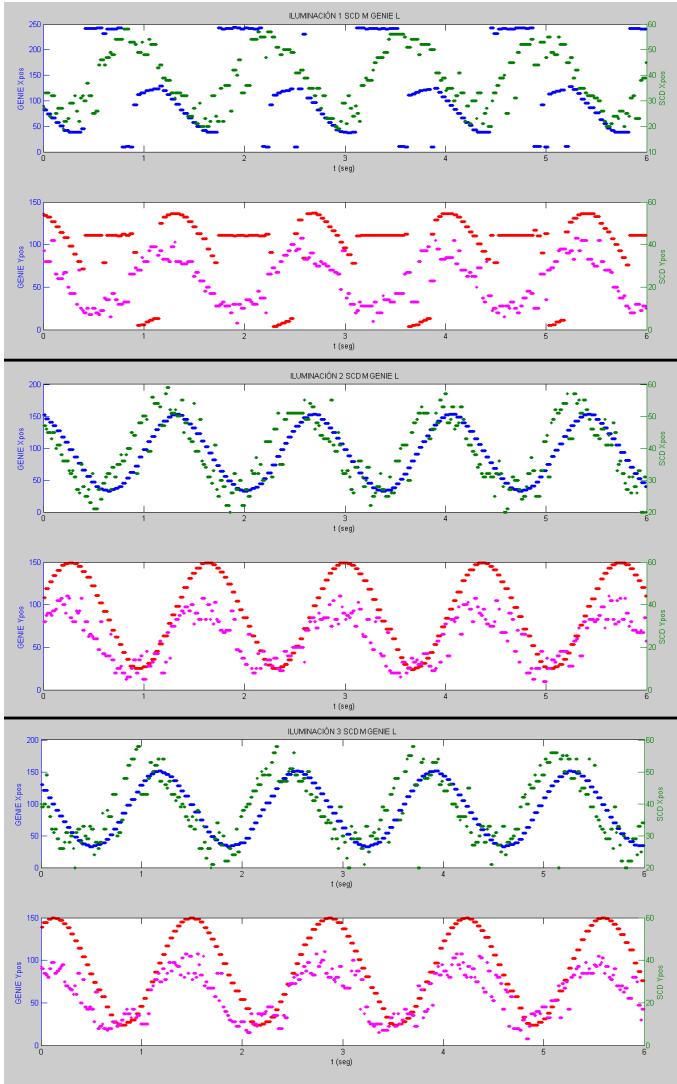


Figura 6.18: Parámetros M y L para el sistema de eventos y el de fotografías, respectivamente. Al igual que en la anterior configuración de L (figura 6.15) en el sistema basado en fotografías, la iluminación1 provoca un comportamiento poco estable, perdiendo la localización del objeto en algunas zonas. El sistema basado en eventos tiene un comportamiento tal como se ha comentado en las imágenes anteriores, estable en las tres configuraciones de iluminación, teniendo un ligero incremento de generación de datos a mayor iluminación.

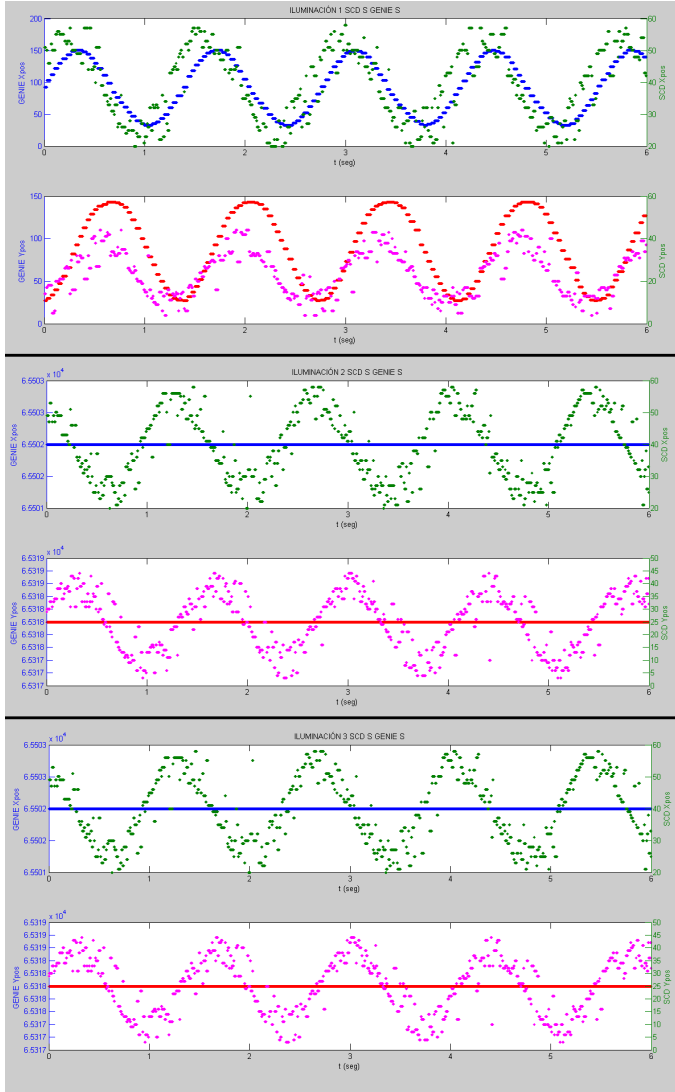


Figura 6.19: Parámetros S para ambos sistemas. Se puede apreciar que para la iluminación 1 y 2 los parámetros con un nivel bajo en la cámara basada en fotografías no se encuentra ningún objeto, sin embargo, con la iluminación 1 tiene un rendimiento óptimo. Por otro lado, la cámara basada en eventos tiene un comportamiento aceptable bajo todas las iluminaciones.

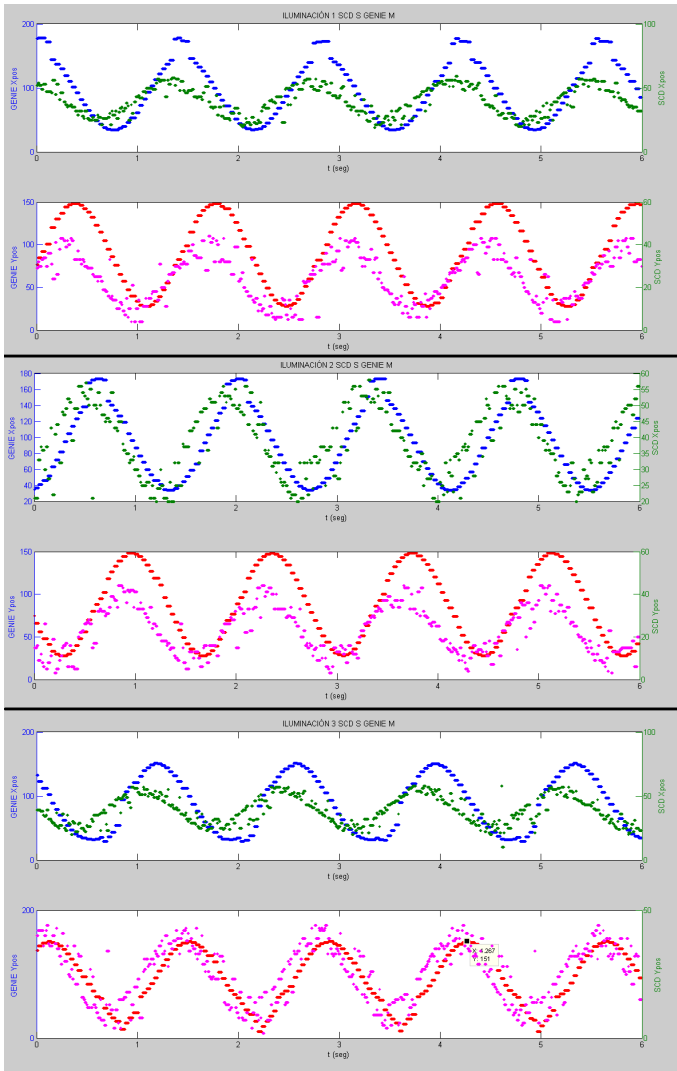


Figura 6.20: Parámetros S y M para el sistema de eventos y el de fotogramas, respectivamente. Se observa que la coordenada X del sistema con fotogramas se distorsiona con los parámetros configurados y con la iluminación1, de forma ligeramente diferente a las anteriores configuraciones. Esto es debido a que la luz ambiental ha variado ligeramente, provocando un resultado diferente en ambos ejes X e Y, pero más acentuado en el X por la incidencia de la luz ambiental.

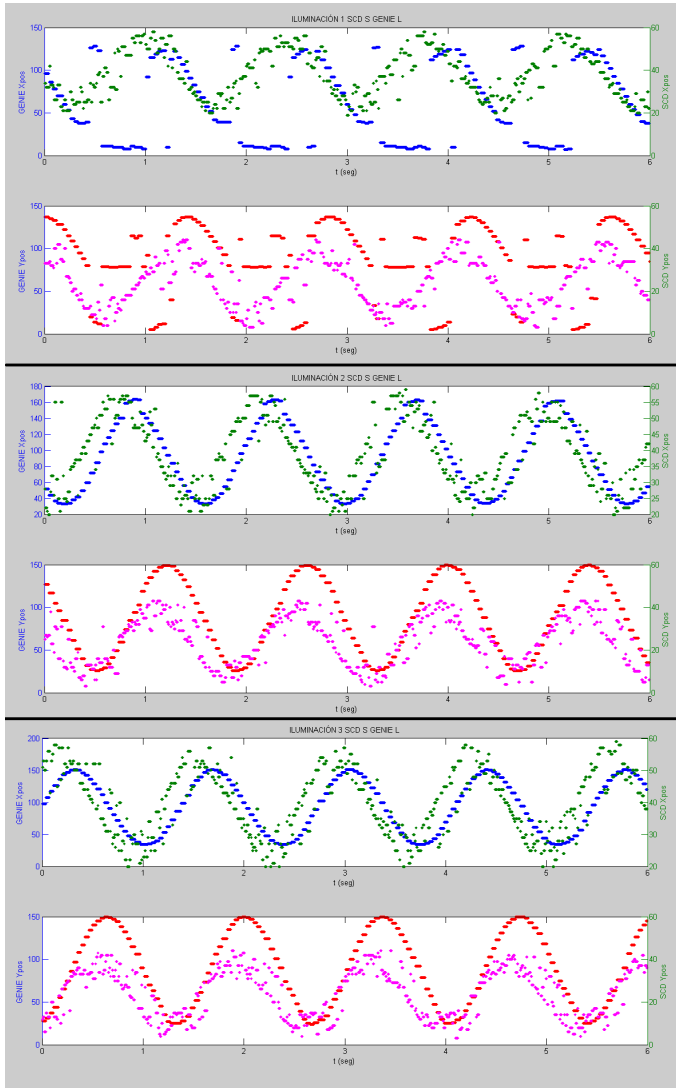


Figura 6.21: Resultado de los datos producidos por ambos sistemas de visión, eventos y fotogramas con los parámetros S y L, respectivamente. Se observa una mayor dispersión de los datos ante la iluminación1 que con el resto de iluminaciones, sin embargo, sigue siendo más rápida la respuesta de la cámara basada en eventos. Puede apreciarse también que la cámara basada en eventos tiene una mejor definición de la trayectoria con la iluminación3. Además, la perturbación de la luz ambiental en la iluminación1 en el sistema basado en fotogramas afecta el seguimiento, siendo ligeramente diferente a las anteriores configuraciones L de este sistema debido a las pequeñas variaciones de iluminación no controlada, demostrando que el sistema basado en eventos además de ser más rápido en la respuesta también es más estable ante cambios de iluminación no controlada.

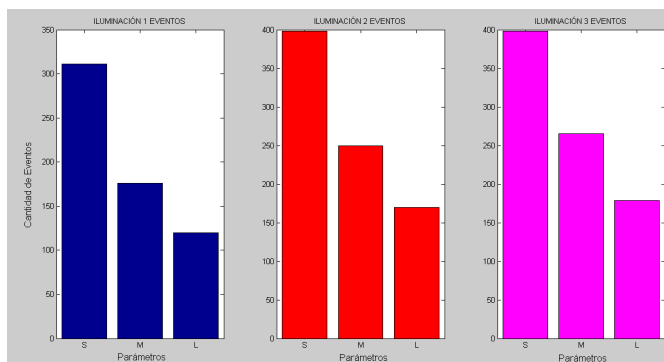


Figura 6.22: Resultados de la sensibilidad de iluminación de la cámara SCD ante cambios de iluminación controlada. Se grafica el total de eventos capturados por el mismo lapso de tiempo bajo las tres diferentes iluminaciones controlables. Como es posible apreciar, el cambio en el resultado para una misma iluminación es el óptimo esperado para el filtro, y la estabilidad del resultado de la cámara es adecuado, pues se ve poco afectado por el cambio en la iluminación, variando menos de 40 eventos por cada configuración de parámetros y las diferentes iluminaciones.

En cuanto al sistema basado en eventos SCD el comportamiento para el mismo conjunto de parámetros es muy similar sin importar la iluminación. No obstante, no es posible concluir que el sistema de eventos es inmune a los cambios de iluminación pues hay ligeras diferencias, pero sí que es mucho menos sensible que un sistema de visión convencional basado en fotografías. En la figura 6.22 se muestran la cantidad de eventos producidos bajo las tres parametrizaciones con las tres iluminaciones. Se puede observar un incremento en la tasa de producción de eventos conforme se pasa de menor intensidad de luz con la iluminación1 a mayor intensidad de luz con la iluminación3. Esto es producido principalmente por el aumento de contraste del objeto con su fondo debido a una mayor intensidad de luz incidente en la escena.

Para poder tener una idea más clara de la variabilidad y repetibilidad de resultados y su sensibilidad ante cambios de iluminación no controlados se repitieron las pruebas en diferentes momentos con las

mismas parametrizaciones y con las mismas fuentes de iluminación. Los resultados se muestran en las figuras 6.23 y 6.24.

Si se observa detenidamente la figura 6.23 se puede apreciar como en el TEST1 y TEST2 existe una pequeña distorsión de la onda senoidal en los mínimos, la cual desaparece en el TEST3. No obstante, en este test la parte superior de la onda senoidal se ve deformada alcanzando valores máximos mayores que en los dos test previos.

En la figura 6.24 se aprecia poca variabilidad en la cantidad de eventos generados captados en diferentes momentos para una misma parametrización. Aunque existe una pequeña variabilidad, la diferencia es despreciable con un muy buen desempeño en cuanto a repetibilidad.

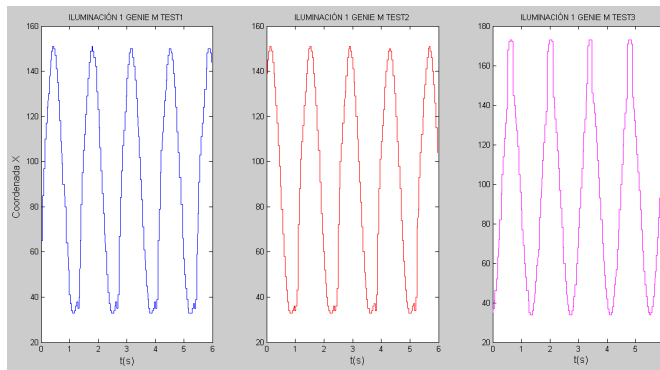


Figura 6.23: Diferencias para los mismos parámetros de filtrado de la cámara basada en fotogramas. El eje X representa el cambio de posición del objeto en la coordenada X, mientras que el eje Y representa el tiempo en segundos. Se observa que este tipo de sistemas de visión es mucho más sensible a los cambios de iluminación ambiental, lo que le hace ser menos repetitivo, y hasta inestable si el ajuste de los parámetros no es el correcto, como en figuras anteriores se ha comentado. El patrón para un mismo movimiento fijo del objeto se observa distorsionado, debido a que las pruebas se hicieron bajo dos fuentes de luz controladas, pero a la intemperie, por lo que la luz del ambiente afecta el resultado.

En términos generales, se puede ver un adelanto en la tendencia de la señal senoidal que reporta la cámara SCD en relación con el que reporta la cámara GENIE. Esta diferencia de tiempo es superior a los 100 ms, y aunque es un tiempo relativamente corto, es suficiente para poder ver un desempeño diferente en el mecanismo robótico.

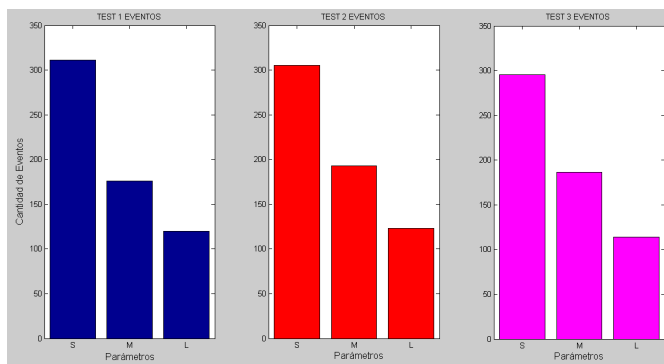


Figura 6.24: Resultados de la sensibilidad de iluminación de la cámara SCD ante cambios de iluminación no controlada. En esta figura se observan tres gráficas del total de eventos generados por la cámara SCD con los tres conjuntos de parámetros de caracterización S, M, L tomadas en diferentes instantes, con diferentes condiciones de iluminación no controlada. Se puede analizar fácilmente que la variabilidad de resultados es casi despreciable, acentuando una de las principales características de una cámara basada en eventos: la baja sensibilidad a cambios de iluminación ambientales.

En la tabla 6.2 se resume el comportamiento comparativo entre la respuesta del sistema basado en eventos SCD y el basado en fotogramas GENIE. Se hace un balance general de todos los resultados con las tres distintas parametrizaciones, y además se presenta por separado los resultados por cada parametrización. El valor de adelanto de reporte de la cámara SCD con respecto a la genie es en promedio de 100 milisegundos para el total de las pruebas realizadas. Para las pruebas hechas con la parametrización menos estricta, S, los valores pueden ser hasta 10 milisegundos más rápidos, aunque se debe considerar que se sacrificará tiempo de respuesta por cantidad de datos producidos. La

	S	M	L	Global
Promedio(ms)	109.6744	99.6305	88.3971	98.0125
Desviación(ms)	98.0585	90.3480	112.4931	97.7256
Máximo(ms)	331.1195	300.1219	333.6380	333.6380
Mínimo(ms)	-136.8033	-177.5677	-276.0221	-276.0221

Tabla 6.2: Resumen comparativo del adelanto en respuesta entre el sistema de visión SCD y el sistema de visión GENIE.

aplicación final de implementación definirá qué debe ser prioritario, y en base a esto hacer un balance de la sintonización del filtrado de eventos.

6.7 Resumen

En las figuras 6.10 a 6.24 se ha mostrado el comportamiento de un sistema de visión que convencionalmente se utiliza en la industria como el basado en fotogramas, comparando a su vez con el sistema de visión artificial basado en eventos.

Se pueden obtener varios análisis objetivos considerando los resultados arrojados. La primera que puede considerarse como ventaja de los sistemas basados en eventos es la rapidez con la que pueden reportar un movimiento: en el sistema planteado, utilizando equipo 100 % industrial, el sistema basado en eventos SCD alcanzaba a reportar en promedio 100 milisegundos antes que un sistema basado en fotogramas GENIE, tiempo que puede ser de consideración en tareas de seguimiento o *tracking*. Aunque cabe mencionar que los datos producidos por la cámara basada en eventos requieren de un pre-procesamiento, debido a la cantidad de ruido y datos dispersos que generan para poder de esta manera limpiar el objeto al que seguir o realizar el tracking. Este pre-pocesamiento de memorización de posiciones de píxeles muertos o espurios resulta útil para poder reducir la dispersión y píxeles cons-

tantemente activos en este tipo de sistemas de visión. Sin embargo, debe tomarse en cuenta que este pre-procesamiento puede afectar a la resolución de los objetos que capta la cámara, pues elimina direcciones del plano XY y cualquier evento que se produzca en esta posición será descartado.

Con respecto a la aplicación del algoritmo LDSI, se puede concluir que los parámetros elegidos cumplen con su nivel de restricción de generación de eventos, cumple con el cometido de ser más o menos estricto en la generación de eventos, o capacidades de filtrado. Todo ello se cumple también aun cuando se cambia la iluminación controlada o hay una variación de la luz ambiental. Por otra parte, para el sistema basado en fotogramas, la iluminación juega un papel fundamental, ya que dependiendo de su incidencia, el comportamiento del sistema era significativamente diferente, perdiendo en momentos, o de manera constante la posición del objeto. Situación que a priori era conocida, pues la iluminación en este tipo de sistemas usualmente toma un alto porcentaje de tiempo de diseño y desarrollo al momento de implementarlos en proyectos de automatización industrial que incorporan sistemas de inspección visual artificial.

En resumen, los sistemas de visión basados en eventos son menos susceptibles a los cambios de iluminación controlados y no controlados, siendo más rápidos en la respuesta. A pesar de ello, se debe considerar que los datos producidos por sistemas de eventos presentan una variabilidad en la respuesta y mayor dispersión de información que puede ser considerada como ruido, que al ser tratada con algoritmos como el LDSI puede resultar interesante para aplicaciones industriales de tracking o seguimiento.

Agradecimiento especial:

Las pruebas aquí presentadas fueron posibles gracias al apoyo del profesor Fernando Pardo de la universidad de Valencia al ceder la cámara basada en eventos. Además, agradezco el apoyo en el montaje y construcción del sistema descrito a Jorge Samaniego, Alejandro Cortina, Marcos García, Taras Iakymchuck y al profesor Antonio Vicente Martínez.

Capítulo 7

Conclusiones

Retomando los objetivos fijados al inicio de este documento, se presentan las conclusiones basadas en los tres ejes principales del desarrollo de la tesis: la codificación y conversión de eventos el filtrado y tratamiento de eventos y el uso de sistemas basados en eventos en la automatización industrial.

7.1 Codificación y conversión de eventos

En esta tesis se han presentado distintas propuestas para codificar sensores que no son basados en eventos y que, con ligeras modificaciones y hardware sencillo, pueden convertirse en eventos, utilizando estos datos para el posterior procesado como si fueran dispositivos nativos basados en eventos. En el capítulo 3 se expusieron y demostraron distintas aplicaciones de codificación de eventos para sensores que no producen eventos.

Fue posible reducir la cantidad de datos generados de manera significativa con respecto a una adquisición tradicional analógica o digital. Es importante tener en cuenta que el desempeño de la medición de los sensores aquí expuestos, y la cantidad de datos generados pueden verse afectados por la calidad de lecturas de datos del sensor, la estabilidad y resolución de la señal que el dispositivo no basado en eventos puede producir. Un algoritmo de filtrado o estabilización podría aumentar la reducción de los datos transmitidos y así obtener mejores resultados.

Aunque depende de la codificación de generación de eventos para dispositivos convencionales, no necesariamente se reducirá la cantidad de datos generados puesto que también se plantearon codificaciones para producir información redundante de una misma medida. Esto no es una desventaja, sino que depende de la utilidad que le dará la aplicación final para discernir si son útiles. Una alta producción de datos sin filtrar usualmente es interesante para utilizar en algoritmos de post-procesamiento para detección, clasificación y en general, aprendizaje automático. Este resultado sugiere que se deben realizar ajustes específicos en la codificación de eventos para cada tipo de sensor.

También se mostró en el capítulo 3 que la implementación de un algoritmo de codificación basado en eventos para sensores convencionales requiere un coste reducido tanto en hardware como en software. La codificación propuesta proporciona datos de eventos listos para ser procesados por redes neuronales *spiking* y otros sistemas neuromórficos que pueden proporcionar resultados eficientes en múltiples aplicaciones, como AER. En este caso, se describieron sensores ampliamente utilizados en robótica móvil como la distancia, la brújula magnética y una cámara de visión. Los resultados obtenidos invitan al uso de información sensorial combinada (visión, distancia, orientación, etc.) en los sistemas neuromórficos de forma que se puedan utilizar múltiples fuentes de información para la toma de decisiones avanzadas y el procesamiento de la información. Todo esto a un costo computacional

reducido, bajo consumo de energía y reducción del tamaño para el hardware, elementos clave en los sistemas móviles.

7.2 Filtrado y tratamiento de eventos

En el capítulo 4 se propuso un filtro bio-inspirado que utiliza datos producidos por sistemas basados en eventos. El algoritmo LDSI tiene la función de ser un filtro configurable y parametrizable, que reduce la cantidad de datos manteniendo la misma información. El nivel de filtrado no solo depende de los valores de los parámetros configurados, sino también de la interacción entre ellos y los tipos de datos de entrada. Se demostró su efectividad en datos producidos por cámaras basadas en eventos, con escenas rápidas, cambio de iluminación, o iluminación incontrolable, elementos que a un sistema de visión convencional basado en fotogramas lo hacen poco robusto. Por esa razón, el algoritmo propuesto LDSI puede ser configurado para adaptarse a diferentes situaciones ambientales, facilitando lo que se quiere medir u observar, y mejorando la información (eventos) que proviene del sensor.

El filtro fue probado tanto con conjuntos de datos previamente grabados como directamente con una cámara basada en eventos o, mejor dicho, de manera *offline* y *online*. *Offline* se hizo para mostrar su repetibilidad y su comportamiento con diversas configuraciones en sus parámetros; y de manera *online* para comprobar su compatibilidad con dispositivos basados en eventos en tiempo real. Se lograron grandes ratios de reducción de datos manteniendo la información de la escena. Los datos producidos pueden alcanzar una reducción de hasta el 30 % en comparación con la captura generada basada en eventos sin filtrar. Este valor puede ser menor dependiendo de las características de la escena que se quieren conservar, pues con una parametrización muy estricta en cuanto al filtrado provocan una atenuación y hasta eliminación de objetos en la escena.

La implementación hardware del algoritmo LDSI muestra que se requiere un bajo uso de recursos para el filtrado, siendo una opción para el procesamiento de sensores basados en eventos de manera centralizada, así como el filtrado embebido en el propio dispositivo generador. Esto es posible gracias al diseño de LDSI, concebido como un bloque de procesamiento intermedio, totalmente compatible con los datos de entrada de un sensor basado en eventos y sirviendo de información a otros dispositivos en tiempos muy cortos, que dependiendo de la aplicación pueden ser despreciables.

Por lo general, las cámaras basadas en eventos producen muchos menos datos que las cámaras convencionales basadas en fotogramas y con el algoritmo propuesto, la reducción es aún mayor. Esta cualidad es muy importante cuando los datos se transmiten a otros sistemas de procesamiento, debido a que normalmente se comparten los canales de transmisión de datos con otros dispositivos y el ancho de banda se puede saturar, perjudicando el rendimiento del sistema.

Con la necesidad de poder evaluar el rendimiento del algoritmo LDSI se diseñaron un par de algoritmos que nos arrojan un índice numérico que pueda servir como valor objetivo comparativo del nivel de filtrado que se obtiene con LDSI. Los algoritmos diseñados fueron llamados ENM y ESM. Estos algoritmos ayudan a encontrar un índice numérico para comparar la cantidad de ruido entre distintas grabaciones para una misma escena, y encontrar otro índice numérico que proporciona una aproximación de qué tan similares son los bordes de los objetos en escena entre dos o más grabaciones. El motivo de requerir dos índices es para contrarrestar el efecto de la eliminación de o atenuación de objetos en escena, cuando el filtro ha sido parametrizado de manera muy estricta, dejando un índice bajo de eventos considerados como ruido, pero la grabación muestra que se ha perdido información relevante. Idealmente, el valor arrojado por ENM debe ser lo menor posible y el valor de ESM lo más cercano al original. Ambos algoritmos son novedosos en el campo del tratamiento de eventos, pues existen pocos

trabajos dedicados a la medición del ruido en datos generados por sistemas basados en eventos.

7.3 Sistemas de eventos en la Automatización Industrial

La integración de nuevos dispositivos en la industria es una necesidad. Diversos nichos de mercado donde la personalización de los productos y servicios, además de la adecuación a las tendencias pueden marcar la diferencia entre ser competitivo o no. Las tareas en un sistema de automatización flexible requieren de interfaces de comunicación de alto rendimiento, por lo que dispositivos lentos o que requieren un ancho de banda son descartados para nuevas instalaciones. En el capítulo 5 se hizo un profundo análisis de dos de los protocolos de comunicaciones, Ethernet/IP y POWERLINK, con una sólida trayectoria en aplicaciones en donde se puede explotar el uso de dispositivos basadas en eventos como lo es la retroalimentación del control de movimiento y *visual servoing*. Ambos protocolos son robustos, con altas prestaciones y aptos para la codificación de eventos entre sus tramas. No obstante, pequeños detalles hacen que para las pruebas presentadas en este trabajo se haya optado por POWERLINK como protocolo de comunicación a implementar. Dos factores hacen más interesante a POWERLINK que a Ethernet/IP para el desarrollo de nuevos dispositivos: el hecho de que POWERLINK sea un protocolo abierto y estándar, y la posibilidad de trabajar de manera síncrona y asíncrona.

Por otra parte, la incorporación de sistemas neuromórficos, que traten de emular o que se basen en las cualidades de los sistemas orgánicos, como bajo consumo de energía, capacidad de procesamiento y almacenamiento, es y ha sido interesante implementar en cualquier campo. Sin embargo, ha sido poco explorada la incorporación a la industria de procesos y manufactura tal como se expone en el capítulo 6. En este trabajo se realizaron pruebas con equipos industriales en sinergia con equipos basados en eventos. Adecuando estos sistemas a los entornos

de trabajo industrial en donde el desempeño tiempo real es prioritario. Dos de los elementos de un sistema de visión convencional que se pretendían comparar en relación a una cámara basada en eventos fueron la sensibilidad a la iluminación y la velocidad de respuesta.

La iluminación juega un papel principal en el diseño y configuración de sistemas de visión convencionales basados en fotogramas, y en las pruebas aquí realizadas quedó confirmada. También se demostró mayor tolerancia de los dispositivos basados en eventos a trabajar con diferentes iluminaciones y menor sensibilidad a cambios de iluminación no controlada.

La velocidad de respuesta también fue superada en comparación con los sistemas basados en fotogramas. Se alcanzaron valores superiores a los 100 milisegundos de diferencia de reacción entre ambos sistemas de visión. Sin embargo, por la propia tecnología de las cámaras basadas en eventos, se requiere de un movimiento relativo entre la cámara y el objeto en la escena, de lo contrario no se producirán eventos. En una situación de *tracking*, como la presentada en este trabajo, la posición del objeto se perdería en caso de que el movimiento deje de producirse. En un entorno industrial esto no suele ocurrir, pues los objetos normalmente están en constante movimiento.

Con base a los resultados, los sistemas neuromórficos basados en eventos pueden ser sustitutos de los sistemas de visión convencionales para aplicaciones de seguimiento, pick&place y conteo, proporcionando mejor capacidad de operación en el entorno industrial que los sistemas convencionales. Es necesario tener claros los alcances de la tecnología y los posibles procesamientos adicionales que requiere cualquiera de las aplicaciones antes mencionadas, usualmente convergentes con el campo de *visual servoing* y la inspección visual.

7.4 Conclusiones Generales

La *Respuesta en Tiempo Real* es uno de los paradigmas que el sector industrial de la automatización lleva décadas tratando de alcanzar y que ha evolucionado acortando sus límites conforme las tecnologías y procesadores han avanzado. Los sistemas basados en eventos son tecnologías interesantes a aplicar en el sector de la industria de la automatización como una posible solución ante situaciones que requieren tiempos de respuesta cortos, además de que poseen de bajo consumo de energía, alta capacidad de procesamiento y sencillez. Los hacen sustitutos de diversos dispositivos convencionales que se utilizan en la actualidad.

En este trabajo se mostró cómo es posible convertir un sensor convencional no basado en eventos a uno basado en eventos con una sencilla modificación de hardware y software. El resultado de esto podría ser fácilmente procesado por algoritmos de procesamiento basado en eventos, como el algoritmo LDSI aquí también propuesto, en cuanto a la reducción de datos manteniendo la misma información, factor clave al momento de transmitir en una red industrial con objetivos de desempeño en tiempo real.

Se ha demostrado como un sistema basado en eventos es más rápido, repetitivo y menos sensible a los cambios de iluminación que un sistema de visión convencional basado en fotogramas. Con esto se trata de atenuar la importancia que juega la iluminación al momento de diseñar e incorporar un sistema de visión artificial en un entorno industrial, en donde el tiempo que se dedica a la selección de la iluminación y el sistema de control de la misma puede ser mayor al tiempo que se invierte en las demás tareas que a priori parecen más complejas como la programación e instalación del sistema.

Si bien es importante tener claros los objetivos de la aplicación que se desea implementar, generalmente los sistemas basados en eventos,

como los que se han utilizado en este trabajo, tendrán mejor cabida en aplicaciones de *tracking*, conteo y en general de movimiento. Sobre todo, en aplicaciones donde la respuesta en tiempo real sea una prioridad, con iluminación controlada variable y con un rango amplio de fuentes de iluminación no controlada.

El resultado de esta implementación con algoritmos de procesamiento de eventos abrió la oportunidad de participar y ganar el concurso ETHERNET-POWERLINK award 2016/2017 [1] organizado por instituciones y empresas del sector de la automatización industrial. Esto es la muestra de que el sector cree que la incorporación de este tipo de tecnologías basadas en eventos puede ser de gran beneficio para la industria.

Bibliografía

- [1] Inf. téc. Ethernet Powerlink Organization, 2017. URL: <https://www.ethernet-powerlink.org/award>.
- [2] H. H. Goldstine y A. Goldstine. “The Electronic Numerical Integrator and Computer (ENIAC)”. En: *IEEE Annals of the History of Computing* 18.1 (Spring de 1996), págs. 10-16. ISSN: 1058-6180. DOI: 10.1109/85.476557.
- [3] A. W. Burks y E. S. Davidson. “Introduction to "The ENIAC"”. En: *Proceedings of the IEEE* 87.6 (jun. de 1999), págs. 1028-1030. ISSN: 0018-9219. DOI: 10.1109/JPROC.1999.763315.
- [4] M. Kruusmaa y col. “FILOSE for Svenning: A Flow Sensing Bioinspired Robot”. En: *IEEE Robotics Automation Magazine* 21.3 (sep. de 2014), págs. 51-62. ISSN: 1070-9932. DOI: 10.1109/MRA.2014.2322287.
- [5] U. Scarfogliero, C. Stefanini y P. Dario. “A bioinspired concept for high efficiency locomotion in micro robots: the jumping Ro-

- bot Grillo”. En: *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. Mayo de 2006, págs. 4037-4042. DOI: 10.1109/ROBOT.2006.1642322.
- [6] W. R. Provancher, S. I. Jensen-Segal y M. A. Fehlberg. “ROCR: An Energy-Efficient Dynamic Wall-Climbing Robot”. En: *IEEE/ASME Transactions on Mechatronics* 16.5 (oct. de 2011), págs. 897-906. ISSN: 1083-4435. DOI: 10.1109/TMECH.2010.2053379.
- [7] V. Chan, S. Liu y A. van Schaik. “AER EAR: A Matched Silicon Cochlea Pair With Address Event Representation Interface”. En: *IEEE Transactions on Circuits and Systems I: Regular Papers* 54.1 (ene. de 2007), págs. 48-59. ISSN: 1549-8328. DOI: 10.1109/TCSI.2006.887979.
- [8] P. Calegari y col. “Genetic approach to radio network optimization for mobile systems”. En: *1997 IEEE 47th Vehicular Technology Conference. Technology in Motion*. Vol. 2. Mayo de 1997, 755-759 vol.2. DOI: 10.1109/VETEC.1997.600430.
- [9] Trevor Lamb. *Evolution of Phototransduction, Vertebrate Photoreceptors and Retina*. Inf. téc. The Organization of the Retina y Visual System, 2018. URL: www.ncbi.nlm.nih.gov/pmc/articles/PMC3143066/.
- [10] D. Betancourt y C. del Rio. “Study of the human eye working principle: an impressive high angular resolution system with simple array detectors”. En: *Fourth IEEE Workshop on Sensor Array and Multichannel Processing, 2006*. Jul. de 2006, págs. 93-97. DOI: 10.1109/SAM.2006.1706098.
- [11] Suzanaerculano-Houzel. “The human brain in numbers: a linearly scaled-up primate brain”. En: *Frontiers in Human Neu-*

-
- rosience* 3 (2009), pág. 31. ISSN: 1662-5161. DOI: 10.3389/neuro.09.031.2009. URL: <https://www.frontiersin.org/article/10.3389/neuro.09.031.2009>.
- [12] Wikimedia Commons. *Biological neuron structure*. 2017. URL: https://en.wikipedia.org/wiki/File:Neuron-no_labels2.png.
- [13] Allan L Hodgkin y Andrew F Huxley. “The dual effect of membrane potential on sodium conductance in the giant axon of *Loligo*”. En: *The Journal of physiology* 116.4 (1952), pág. 497.
- [14] C. Mead. “Neuromorphic electronic systems”. En: *Proceedings of the IEEE* 78.10 (oct. de 1990), págs. 1629-1636. ISSN: 0018-9219. DOI: 10.1109/5.58356.
- [15] Tobi Delbrück y Shih-Chii Liu. “A silicon early visual system as a model animal”. En: *Vision Research* 44.17 (2004), págs. 2083-2089. ISSN: 0042-6989. DOI: <https://doi.org/10.1016/j.visres.2004.03.021>. URL: <http://www.sciencedirect.com/science/article/pii/S0042698904001531>.
- [16] M. Mahowald. “Computation and Neural Systems”. Tesis doct. California Institute of Technology, 1992.
- [17] T. Delbruck y R. Berner. “Temporal contrast AER pixel with 0.3contrast event threshold”. En: *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*. Mayo de 2010, págs. 2442-2445. DOI: 10.1109/ISCAS.2010.5537153.
- [18] F. Pardo, J. A. Boluda y F. Vegara. “Selective Change Driven Vision Sensor With Continuous-Time Logarithmic Photoreceptor and Winner-Take-All Circuit for Pixel Selection”.

- En: *IEEE Journal of Solid-State Circuits* 50.3 (mar. de 2015), págs. 786-798. ISSN: 0018-9200. DOI: 10.1109/JSSC.2014.2386899.
- [19] R. Serrano-Gotarredona y col. “CAVIAR: A 45k Neuron, 5M Synapse, 12G Connects/s AER Hardware Sensory Learning Actuating System for High-Speed Visual Object Recognition and Tracking”. En: *IEEE Transactions on Neural Networks* 20.9 (sep. de 2009), págs. 1417-1438. ISSN: 1045-9227. DOI: 10.1109/TNN.2009.2023653.
- [20] E. M. Izhikevich. “Simple model of spiking neurons”. En: *IEEE Transactions on Neural Networks* 14.6 (nov. de 2003), págs. 1569-1572. ISSN: 1045-9227. DOI: 10.1109/TNN.2003.820440.
- [21] R. Serrano-Gotarredona y col. “An arbitrary kernel convolution AER-transceiver chip for real-time image filtering”. En: *2006 IEEE International Symposium on Circuits and Systems*. Mayo de 2006, 4 pp.-. DOI: 10.1109/ISCAS.2006.1693292.
- [22] M. Rivas-Perez y col. “AER spike-processing filter simulator: Implementation of an AER simulator based on cellular automata”. En: *Proceedings of the International Conference on Signal Processing and Multimedia Applications*. Jul. de 2011, págs. 1-6.
- [23] A. Espínola y col. “Robust vision system to illumination changes in a color-dependent task”. En: *2011 IEEE International Conference on Robotics and Biomimetics*. Dic. de 2011, págs. 521-526. DOI: 10.1109/ROBIO.2011.6181339.
- [24] W. K. Lin y col. “LED strobe lighting for machine vision inspection”. En: *2013 International Symposium on Next-Generation*

-
- Electronics*. Feb. de 2013, págs. 345-346. DOI: 10.1109/ISNE.2013.6512362.
- [25] H. Kim y col. “Color mixing and random search for optimal illumination in machine vision”. En: *Proceedings of the 2013 IEEE/SICE International Symposium on System Integration*. Dic. de 2013, págs. 907-912. DOI: 10.1109/SII.2013.6776736.
- [26] J. Lazzaro y col. “Silicon auditory processors as computer peripherals”. En: *IEEE Transactions on Neural Networks* 4.3 (mayo de 1993), págs. 523-528. ISSN: 1045-9227. DOI: 10.1109/72.217193.
- [27] J. Lazzaro y J. Wawrzynek. “A multi-sender asynchronous extension to the AER protocol”. En: *Proceedings Sixteenth Conference on Advanced Research in VLSI*. Mar. de 1995, págs. 158-169. DOI: 10.1109/ARVLSI.1995.515618.
- [28] *The Address-Event Representation Communication Protocol*. url: <https://www.ini.uzh.ch/amw/scx/std002.pdf>. 1993 [ONLINE].
- [29] Wolfgang Maass. “Networks of spiking neurons: The third generation of neural network models”. En: *Neural Networks* 10.9 (1997), págs. 1659-1671. ISSN: 0893-6080. DOI: [https://doi.org/10.1016/S0893-6080\(97\)00011-7](https://doi.org/10.1016/S0893-6080(97)00011-7). URL: <http://www.sciencedirect.com/science/article/pii/S0893608097000117>.
- [30] R. Ghosh y col. “Real-time object recognition and orientation estimation using an event-based camera and CNN”. En: *2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings*. Oct. de 2014, págs. 544-547. DOI: 10.1109/BioCAS.2014.6981783.

- [31] J. A. Pérez-Carrasco y col. “Mapping from Frame-Driven to Frame-Free Event-Driven Vision Systems by Low-Rate Rate Coding and Coincidence Processing—Application to Feedforward ConvNets”. En: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.11 (nov. de 2013), págs. 2706-2719. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2013.71.
- [32] T. Serrano-Gotarredona y col. “ConvNets experiments on SpiN-Naker”. En: *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*. Mayo de 2015, págs. 2405-2408. DOI: 10.1109/ISCAS.2015.7169169.
- [33] B. Zhao y col. “Feedforward Categorization on AER Motion Events Using Cortex-Like Features in a Spiking Neural Network”. En: *IEEE Transactions on Neural Networks and Learning Systems* 26.9 (sep. de 2015), págs. 1963-1978. ISSN: 2162-237X. DOI: 10.1109/TNNLS.2014.2362542.
- [34] A. Jimenez-Fernandez y col. “Neuro-inspired system for real-time vision sensor tilt correction”. En: *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*. Mayo de 2010, págs. 1394-1397. DOI: 10.1109/ISCAS.2010.5537271.
- [35] Offnfopt. *Modelo OSI*. Inf. téc. Offnfopt, 2018. URL: https://commons.wikimedia.org/wiki/File:OSI_Model_v1.svg.
- [36] J. D. Decotignie. “Ethernet-Based Real-Time and Industrial Communications”. En: *Proceedings of the IEEE* 93.6 (jun. de 2005), págs. 1102-1117. ISSN: 0018-9219. DOI: 10.1109/JPROC.2005.849721.
- [37] J. Partzsch y col. “Live demonstration: Ethernet communication linking two large-scale neuromorphic systems”. En: *2013*

-
- European Conference on Circuit Theory and Design (ECCTD)*.
Sep. de 2013, págs. 1-1. DOI: 10.1109/ECCTD.2013.6662196.
- [38] D. B. Fasnacht, A. M. Whatley y G. Indiveri. “A serial communication infrastructure for multi-chip address event systems”. En: *2008 IEEE International Symposium on Circuits and Systems*. Mayo de 2008, págs. 648-651. DOI: 10.1109/ISCAS.2008.4541501.
- [39] M. Jablonski, T. Serrano-Gotarredona y B. Linares-Barranco. “High-speed serial interfaces for event-driven neuromorphic systems”. En: *2015 International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP)*. Jun. de 2015, págs. 1-4. DOI: 10.1109/EBCCSP.2015.7300697.
- [40] Michela Nalin. *Industrial Ethernet is now bigger than fieldbuses*. Inf. téc. HMS Network, 2018. URL: <https://www.hms-networks.com/press/2018/02/27/industrial-ethernet-is-now-bigger-than-fieldbuses>.
- [41] L. Winkel. “Real-Time Ethernet in IEC 61784-2 and IEC 61158 series”. En: *2006 4th IEEE International Conference on Industrial Informatics*. Ago. de 2006, págs. 246-250. DOI: 10.1109/INDIN.2006.275788.
- [42] Martin Rostan. *Industrial Ethernet Technologies*. Inf. téc. Ethercat Organization, 2014. URL: https://www.ethercat.org/download/documents/Industrial_Ethernet_Technologies.pdf.
- [43] Richard Zurawski. *Industrial Communication Technology Handbook*. Ed. por 2da Edición. CRC Press, 2014.

- [44] Bogdan M. Wilamowski y J. David Irwin. *Industrial Communication Systems*. 2nd. Boca Raton, FL, USA: CRC Press, Inc., 2011. ISBN: 9781439802816.
- [45] P. Brooks. “Ethernet/IP-industrial protocol”. En: *ETFA 2001. 8th International Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No.01TH8597)*. Vol. 2. Oct. de 2001, 505-514 vol.2. DOI: 10.1109/ETFA.2001.997725.
- [46] Open DeviceNet Vendor Association. *The Common Industrial Protocol*. Inf. téc. ODVA, 2018. URL: <https://www.odva.org>.
- [47] Modbus. *Modbus Specifications and Implementation Guides*. Inf. téc. Modbus Organization, 2018. URL: <http://www.modbus.org/specs.php>.
- [48] EPSG. *Industrial Ethernet Facts*. Inf. téc. Ethernet Powerlink Standarization Group, 2017. URL: <https://www.ethernet-powerlink.org/downloads/industrial-ethernet-facts>.
- [49] J. Conradt y col. “A pencil balancing robot using a pair of AER dynamic vision sensors”. En: *2009 IEEE International Symposium on Circuits and Systems*. Mayo de 2009, págs. 781-784. DOI: 10.1109/ISCAS.2009.5117867.
- [50] A. Linares-Barranco y col. “Using FPGA for visuo-motor control with a silicon retina and a humanoid robot”. En: *2007 IEEE International Symposium on Circuits and Systems*. Mayo de 2007, págs. 1192-1195. DOI: 10.1109/ISCAS.2007.378265.
- [51] T. Delbruck y col. “Human vs. computer slot car racing using an event and frame-based DAVIS vision sensor”. En: *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*.

- Mayo de 2015, págs. 2409-2412. DOI: 10.1109/ISCAS.2015.7169170.
- [52] A. Linares-Barranco y col. “On algorithmic rate-coded AER generation”. En: *IEEE Transactions on Neural Networks* 17.3 (mayo de 2006), págs. 771-788. ISSN: 1045-9227. DOI: 10.1109/TNN.2006.872253.
- [53] R. Benosman y col. “Asynchronous Event-Based Hebbian Epipolar Geometry”. En: *IEEE Transactions on Neural Networks* 22.11 (nov. de 2011), págs. 1723-1734. ISSN: 1045-9227. DOI: 10.1109/TNN.2011.2167239.
- [54] D. G. Chen, A. Bermak y C. Y. Tsui. “A low-complexity image compression algorithm for Address-Event Representation (AER) PWM image sensors”. En: *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*. Mayo de 2011, págs. 2825-2828. DOI: 10.1109/ISCAS.2011.5938193.
- [55] V. Chan, S. C. Liu y A. van Schaik. “AER EAR: A Matched Silicon Cochlea Pair With Address Event Representation Interface”. En: *IEEE Transactions on Circuits and Systems I: Regular Papers* 54.1 (ene. de 2007), págs. 48-59. ISSN: 1549-8328. DOI: 10.1109/TCSI.2006.887979.
- [56] F. Gómez-Rodríguez y col. “ED-Scorbot: A robotic test-bed framework for FPGA-based neuromorphic systems”. En: *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. Jun. de 2016, págs. 237-242. DOI: 10.1109/BIOROB.2016.7523630.
- [57] Fred Rieke y col. *Spikes: Exploring the Neural Code*. MIT Press, 1999. ISBN: 0262181746.

- [58] Wulfram Gerstner y Werner Kistler. *Spiking Neuron Models: An Introduction*. Cambridge University Press, 2002. ISBN: 0521890799.
- [59] Juan Barrios-Avilés. *Ultrasonic and Digital compass*. Inf. téc. Universidad de Valencia, 2016. URL: <https://github.com/juabarav>.
- [60] *JAER Open Source Project*. url: <http://jaer.wiki> ; sourceforge.net. 2007 [ONLINE].
- [61] iniVation AG. *AEDAT file formats*. Inf. téc. Inilabs, 2018. URL: <https://inivation.com/support/software/fileformat/>.
- [62] ARDUCAM. *5MP: OV5642*. Inf. téc. ARDUCAM, 2016. URL: <http://www.arducam.com/camera-modules/5mp-ov5642/>.
- [63] *what-is-qt*. url: <https://www.qt.io/what-is-qt/>. 2015 [ONLINE].
- [64] S. Hutchinson, G. D. Hager y P. I. Corke. “A tutorial on visual servo control”. En: *IEEE Transactions on Robotics and Automation* 12.5 (oct. de 1996), págs. 651-670. ISSN: 1042-296X. DOI: 10.1109/70.538972.
- [65] iniVation AG. *DVS and DAVIS Specifications*. Inf. téc. Inilabs, 2017. URL: <https://inivation.com/wp-content/uploads/2018/06/2018-06-08-DVS-Specifications.pdf>.
- [66] *MNIST DATA BASE*. url: <http://www2.imse-cnm.csic.es/caviar/MNISTDVS> 2016 [ONLINE].
- [67] Xilinx. *Virtex-6 FPGA ML605 Evaluation Kit*. Inf. téc. Xilinx, 2015. URL: <https://www.xilinx.com/products/boards-and-kits/ek-v6-ml605-g.html>.

-
- [68] Teledyne. *Sherlock Vision System Software. User Manual and Embedded User Manual*. Inf. téc. Teledyne DALSA, 2017. URL: <https://www.teledynedalsa.com/imaging/products/vision-systems/software/sherlock/>.
- [69] J. Canny. “A Computational Approach to Edge Detection”. En: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8.6 (nov. de 1986), págs. 679-698. ISSN: 0162-8828. DOI: 10.1109/TPAMI.1986.4767851.
- [70] M. Antonini y col. “Image coding using wavelet transform”. En: *IEEE Transactions on Image Processing* 1.2 (abr. de 1992), págs. 205-220. ISSN: 1057-7149. DOI: 10.1109/83.136597.
- [71] D. P. Huttenlocher, G. A. Klanderman y W. J. Rucklidge. “Comparing images using the Hausdorff distance”. En: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15.9 (sep. de 1993), págs. 850-863. ISSN: 0162-8828. DOI: 10.1109/34.232073.
- [72] S. Geman y D. Geman. “Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images”. En: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-6.6 (nov. de 1984), págs. 721-741. ISSN: 0162-8828. DOI: 10.1109/TPAMI.1984.4767596.
- [73] J. S. Lee. “Digital Image Enhancement and Noise Filtering by Use of Local Statistics”. En: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-2.2 (mar. de 1980), págs. 165-168. ISSN: 0162-8828. DOI: 10.1109/TPAMI.1980.4766994.

- [74] A. J. Bardos y S. J. Sangwine. “Measuring noise in colour images”. En: *IEE Colloquium on Non-Linear Signal and Image Processing (Ref. No. 1998/284)*. Mayo de 1998, págs. 8/1-8/4. DOI: 10.1049/ic:19980443.
- [75] Yeong-Hwa Kim y Jaeheon Lee. “Image feature and noise detection based on statistical hypothesis tests and their applications in noise reduction”. En: *IEEE Transactions on Consumer Electronics* 51.4 (nov. de 2005), págs. 1367-1378. ISSN: 0098-3063. DOI: 10.1109/TCE.2005.1561869.
- [76] Ce Liu y col. “Noise Estimation from a Single Image”. En: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 1. Jun. de 2006, págs. 901-908. DOI: 10.1109/CVPR.2006.207.
- [77] Zhou Wang y A. C. Bovik. “A universal image quality index”. En: *IEEE Signal Processing Letters* 9.3 (mar. de 2002), págs. 81-84. ISSN: 1070-9908. DOI: 10.1109/97.995823.
- [78] T. Serrano-Gotarredona, A. G. Andreou y B. Linares-Barranco. “Programmable 2D image filter for AER vision processing”. En: *Circuits and Systems, 1999. ISCAS '99. Proceedings of the 1999 IEEE International Symposium on*. Vol. 4. Jul. de 1999, 159-162 vol.4. DOI: 10.1109/ISCAS.1999.779966.
- [79] D. Weikersdorfer y J. Conradt. “Event-based particle filtering for robot self-localization”. En: *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. Dic. de 2012, págs. 866-870. DOI: 10.1109/ROBIO.2012.6491077.
- [80] L. A. Camuñas-Mesa y col. “Event-driven stereo vision with orientation filters”. En: *2014 IEEE International Symposium*

-
- on Circuits and Systems (ISCAS)*. Jun. de 2014, págs. 257-260. DOI: 10.1109/ISCAS.2014.6865114.
- [81] Viktor Schiffer. *COMMON INDUSTRIAL PROTOCOL (CIP) AND THE FAMILY OF CIP NETWORKS*. Inf. téc. Ann Arbor, Michigan, USA: ODVA Inc, 2016. URL: <https://www.odva.org/>.
- [82] Raimond Pigan y Mark Metter. *Automating with PROFINET: Industrial Communication Based on Industrial Ethernet*. 2.^a ed. Publicis MCD Verlag, Germany, 2008. ISBN: 978-3895782947.
- [83] EtherCAT. *EtherCAT the Ethernet Fieldbus*. Inf. téc. Ostendsstrasse 196, 90482 Nuremberg, Germany: EtherCAT organisation, 2014. URL: <https://ethercat.org>.
- [84] EPSG. *Ethernet POWERLINK Communication Profile Specification*. Inf. téc. Bonsaiweg 6, 15370 Fredersdorf, Germany: Ethernet Powerlink Standardisation Group - EPSG, 2013. URL: <http://www.ethernet-powerlink.org/>.
- [85] Sercos. *A comparison: Sercos and Ethercat*. Inf. téc. Sercos Organization, 2018. URL: <https://www.sercos.org/downloads/studies/>.
- [86] Profibus Organization. *The difference between Profibus and Profinet*. Inf. téc. Profibus Organization, 2018. URL: <http://us.profinet.com/white-paper/profibus-vs-profinet-comparison-migration-strategies/>.
- [87] ODVA. *Extending EtherNet IPTM to Resource-Constrained*. Inf. téc. ODVA, 2015. URL: <https://www.odva.org/Portals/0/>

Library/Conference/2015_ODVA_Conference_Xu-Brooks_Resource-Constrained-Things.pdf.

- [88] H. Bloch, A. Fay y M. Hoernicke. “Analysis of service-oriented architecture approaches suitable for modular process automation”. En: *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*. Sep. de 2016, págs. 1-8. DOI: 10.1109/ETFA.2016.7733651.
- [89] L. Dürkop, J. Jasperneite y A. Fay. “An analysis of real-time ethernetets with regard to their automatic configuration”. En: *2015 IEEE World Conference on Factory Communication Systems (WFCS)*. Mayo de 2015, págs. 1-8. DOI: 10.1109/WFCS.2015.7160548.
- [90] B. J. Gruenwald. “Integrating factory automation: a new approach”. En: *Fourth Annual Canadian Conference Proceedings., Programmable Control and Automation Technology Conference and Exhibition*. Oct. de 1988, 12A2-1/1-9. DOI: 10.1109/PROCCE.1988.82237.
- [91] CANopen. *CAN-based higher-layer protocols*. Inf. téc. CAN-CIA, 2018. URL: <https://www.can-cia.org/can-knowledge/hlp/higher-layer-protocols/>.
- [92] Ethernet POWERLINK Standarization Group. *POWERLINK NOW IEEE 61158 STANDARD*. Inf. téc. EPSG, 2017. URL: <https://www.ethernet-powerlink.org/news-events/news/news-detail/powerlink-now-ieee-61158-standard>.
- [93] OMRON Corporation. *NJ501*. Inf. téc. OMRON Corporation, 2016. URL: <http://www.ia.omron.com/products/family/3075/>.

- [94] OMRON Corporation. *NJ101*. Inf. téc. OMRON Corporation, 2016. URL: <http://www.ia.omron.com/products/family/3438/>.
- [95] OMRON Corporation. *Sysmac Studio*. Inf. téc. OMRON Corporation, 2016. URL: <https://industrial.omron.es/es/products/sysmac-studio>.
- [96] B&R. *X20 system user's manual v3.35*. Inf. téc. B&R Industrial Automation, 2016. URL: <https://www.br-automation.com/en/products/control-systems/x20-system/compact-cpus/x20cp1382/>.
- [97] B&R. *Automation Studio*. Inf. téc. B&R, 2018. URL: <https://www.br-automation.com/es/productos/software/automation-studio>.
- [98] TP-LINK. *TL-SG3210*. Inf. téc. TP-LINK, 2016. URL: https://www.tp-link.com/es/products/details/cat-39_TL-SG3210.html.
- [99] EPSG. *OPENPOWERLINK*. Inf. téc. EPSG, 2018. URL: <http://openpowerlink.sourceforge.net/web/>.
- [100] ANYBUS. *CompactCom M40 Module - Powerlink. MMA316 Version 2. Network guide, Hardware design guide and Software design guide*. Inf. téc. HMS Industrial Networks, 2017. URL: <https://www.anybus.com/support/file-doc-downloads%5C%5C20/compactcom-40-series-specific/?ordercode=AB6611>.
- [101] ANYBUS. *Powerlink XDD FILE Uploader*. Inf. téc. HMS Industrial Networks, 2018. URL: <https://www.anybus.com/>

docs / librariesprovider7 / default - document - library / software / powerlink - xdd - file - uploader . zip ? sfvrsn = 7836aad6_14.

- [102] B&R. *ACOPOSmicro User's manual v1.20*. Inf. téc. B&R Industrial Automation, 2016. URL: <https://www.br-automation.com/en/products/motion-control/acoposmicro/>.
- [103] B&R. *8LVA three-phase synchronous motors. User's manual v1.0*. Inf. téc. B&R Industrial Automation, 2017. URL: <https://www.br-automation.com/en/products/motion-control/8lva-compact-motor/size-1/8lva13ee030ffgg-0/>.
- [104] Avnet. *ZedBoard (Zynq Evaluation and Development). Hardware Users Guide v2.2*. Inf. téc. Avnet, 2014. URL: <http://zedboard.org/support/documentation/1521>.
- [105] Jose A. Boluda y col. "Selective Change Driven Imaging: A Biomimetic Visual Sensing Strategy". En: *Sensors* 11.11 (2011), págs. 11000-11020. ISSN: 1424-8220. DOI: 10.3390/s111111000. URL: <http://www.mdpi.com/1424-8220/11/11/11000>.
- [106] DALSA. *GV1000 Vision System. Installation manual v4.5*. Inf. téc. Teledyne DALSA, 2016. URL: <https://www.teledynedalsa.com/imaging/products/vision-systems/systems/>.
- [107] DALSA. *GENIE Monochrome Series manual*. Inf. téc. Teledyne DALSA, 2011. URL: <http://www.teledynedalsa.com/imaging/products/cameras/area-scan/genie/CR-GEN3-M640X/>.

- [108] SMC. *VX21 solenoid valve manual*. Inf. téc. SMC Corporation, 2016. URL: <http://www.smcworld.com/products/en/get.do?type=GUIDE%5C&id=VX2-E>.
- [109] Fernando Pardo, Jose A. Boluda y Francisco Vegara. “Random telegraph signal transients in active logarithmic continuous-time vision sensors”. En: *Solid-State Electronics* 114 (2015), págs. 111-114. ISSN: 0038-1101. DOI: <https://doi.org/10.1016/j.sse.2015.08.025>. URL: <http://www.sciencedirect.com/science/article/pii/S0038110115002531>.
- [110] PLCopen. *PLC MOTION CONTROL*. Inf. téc. PLCopen, 2016. URL: http://www.plcopen.org/pages/tc2_motion_control/.
- [111] Juan Barrios. *Videos de SCD y GENIE*. Inf. téc. Universidad de Valencia, 2018. URL: https://www.youtube.com/playlist?list=PLleYDu040SjTaARxjGDGcnRMGTTnJ3KN_.

