



Universitat
de les Illes Balears

Revisión extensa de métodos de detección de obstáculos para entornos genéricos mediante técnicas de visión por computador

Joan P. Company-Corcoles, Alberto Ortiz

Departament
de Ciències Matemàtiques
i Informàtica

TECHNICAL REPORT

Enero / 2019

A-01-2019

Revisión extensa de métodos de detección de obstáculos para entornos genéricos mediante técnicas de visión por computador

Joan P. Company-Corcoles, Alberto Ortiz
Departamento de Ciencias Matemáticas e Informática
Universidad de las Islas Baleares
07122 Palma de Mallorca, España
joanpep.company@uib.es

Resumen—En los últimos años, la utilización de algoritmos basados en visión ha crecido debido a los avances en procesadores, así como a las mejoras en los sensores de imagen. Uno de los tópicos más importantes en robótica móvil es la navegación del robot de forma segura, esto es, sin colisionar con obstáculos. Para conseguir que la tarea de navegación se lleve a cabo de forma satisfactoria es por tanto necesario poder detectar obstáculos y zonas libres del entorno. En este trabajo se revisan y comparan varios métodos de detección de obstáculos y zonas libres basados en visión. Este artículo contempla métodos genéricos de detección de obstáculos, en lugar de centrarse en la detección de un objeto específico o el uso de plataformas determinadas como hacen otros estados del arte existentes.

I. INTRODUCCIÓN

En el ámbito de la robótica móvil, para poder navegar de forma segura por el entorno es necesario poder detectar obstáculos o zonas libres del entorno. En el campo de la navegación, un obstáculo se puede definir como un objeto que obstruye o dificulta el movimiento hacia un lugar específico. Actualmente existen muchos métodos capaces de detectar obstáculos cuyos datos sensoriales provienen de cámaras, cámaras binoculares, escáneres láser, etc. Este trabajo se centra en métodos cuyos datos sensoriales provienen de sistemas basados en visión.

Los métodos de detección de obstáculos basados en visión que se citan en este artículo abarcan diferentes niveles de complejidad. Estos van desde el sistema más simple donde se detecta que hay un obstáculo en la imagen sin conocer su posición ni profundidad, hasta los más complejos que permiten detectar y posicionar múltiples obstáculos, estáticos o dinámicos, y actualizar su posición en un mapa previamente generado. Por otra parte, otra forma de llevar a cabo una tarea de navegación de forma satisfactoria es detectando zonas libres. Algunos de los métodos que se citan en este artículo son capaces de encontrar zonas transitables o seguras basándose en esta suposición. La detección de obstáculos o de zonas libres de ellos es normalmente un paso previo a tareas como las de navegación [1], elaboración de mapas semánticos [2] o el seguimiento de objetos [3].

En los últimos años se han incrementado la cantidad de métodos de detección de obstáculos que detectan peatones,

coches y aceras, es decir, entornos relacionados con automóviles. En la actualidad existen varios estados del arte de detección de obstáculos para coches. Por ejemplo, en [4], se presenta un estado del arte relacionado con la detección de obstáculos no específicos de forma autónoma usando sistemas binoculares en vehículos terrestres, mientras que otros estados del arte se centran en la detección de determinados obstáculos como peatones [5], [6], [7] o vehículos [8], [9]. Cabe destacar que algunos métodos incluidos en estados del arte anteriores han sido también incluidos en este artículo ya que presentan métodos interesantes que pueden utilizarse en plataformas robóticas. A diferencia de los estados de arte citados anteriormente, los cuales se basan en coches, este artículo se centra en los métodos de detección más que en la plataforma robótica y el entorno donde se ejecuta. Por ello en este estado del arte se pueden encontrar múltiples tipos de plataformas robóticas entre las que cabe destacar aéreas, terrestres y submarinas.

Cada uno de los estados del arte de detección de obstáculos tienen su propia forma de agrupar los métodos. Por ejemplo, en [10] se utiliza una clasificación basada en métodos de divergencia de flujo, estructura del movimiento, segmentación del plano de tierra y métodos biológicamente inspirados. A diferencia del anterior estado del arte, en este trabajo se ha optado por una clasificación diferente, que parte del hecho diferenciador de disponer de la distancia a los obstáculos o no. Por ello, el primer nivel de nuestra clasificación diferencia entre métodos que utilizan esta información y métodos que no la usan. El siguiente nivel hace referencia al sensor. En sistemas de visión este sensor puede ser una cámara monocular o una cámara binocular, donde cada tipo de sensor tiene métodos específicos de extracción de la profundidad. Siguiendo esta línea, en el apartado de las cámaras binoculares se explican los diferentes métodos de generación de mapas existentes capaces de representar diferentes obstáculos, mientras que en la parte de sistemas monoculares, se clasifican en métodos que calculan el tiempo hasta la colisión (TTC), los que utilizan mapas y otros sistemas. Al respecto de los métodos que no estiman la distancia a los obstáculos, se pueden encontrar métodos de aprendizaje automático, de mapeo de perspectiva inversa (IPM), de segmentación de la imagen basado en su apariencia y otros métodos. En la figura I se puede encontrar una visión general de la

clasificación utilizada en este artículo.

Este trabajo no pretende una revisión exhaustiva de todos los métodos publicados, debido a que su número es inmenso. Nos restringiremos, por tanto, a los más relevantes. Tampoco se consideran los métodos basados en el uso de marcadores visuales tales como ArUco [11] o ARTag [12]. Como por ejemplo en [13] y [14], ya que los marcadores se colocan de forma artificial en el entorno.

Un conjunto de métodos que se debe destacar en esta introducción debido a su importancia son los trabajos conocidos como biológicamente inspirados. Estos métodos de navegación y detección de obstáculos se basan en el comportamiento de insectos y animales, utilizando técnicas visuales de baja complejidad, en lugar de manejar un gran conocimiento del entorno. Los métodos biológicamente inspirados pueden tener varios propósitos, como la detección de obstáculos frontales, auto-estabilización, fijación de objetos, cálculo de la velocidad utilizando una evaluación del suelo y centrado de la plataforma. En este estado del arte no estarán agrupados en una sola categoría, ya que lo forman varios métodos diferentes.

Dentro de los trabajos biológicamente inspirados existen métodos para detectar obstáculos mediante visión basados en calcular lo que se conoce como flujo óptico (OF). El OF es uno de los conceptos más utilizados desde los orígenes de la visión por computador. Los trabajos originales se remontan al año 1981 con autores como Lucas-Kanade [15] y Horn y Schunck [16]. Desde entonces, se ha realizado mucha investigación entorno al OF. En este trabajo tampoco hay un apartado específico reservado a esta técnica, ya que el OF es utilizado por varias familias de métodos descritas en este estado del arte tales como el flujo binocular, métodos de seguimiento, de segmentación de objetos, de TTC, etc. Debido a su importancia, a continuación se resumen sus características más importantes. El OF es una técnica de detección y evitación de obstáculos basada en insectos y pájaros [17]. Mediante un campo vectorial de dos dimensiones es capaz de describir el movimiento aparente de cada píxel entre imágenes de una escena dinámica de tres dimensiones capturada en diferentes instantes de tiempo. Para poder conseguir una buena estimación del flujo óptico se han de cumplir una serie de condiciones entre las cuales se destacan que la luz entre las dos imágenes ha de mantenerse estable, evitando su uso en lugares donde haya parpadeos como los producidos por los fluorescentes, iluminación uniforme en los objetos y que no haya aparición de sombras o especularidades. Todos estos factores pueden provocar una mala estimación del movimiento aparente de cada píxel. El OF es una metodología oportuna para la detección de obstáculos basada en visión utilizada en robots. En [18] se detalla una investigación de las formas de calcular OF, mientras que en [19] se analizan las diferentes técnicas de OF existentes utilizadas en la navegación de robots. La popularidad del OF como único elemento para detectar obstáculos ha sido relevada por otras técnicas como las basadas en disparidad, aprendizaje automático. Sin embargo, el número de publicaciones de diferentes métodos actuales combinados con OF se ha incrementado.

Este artículo continúa con las siguientes secciones. La

sección II describe y compara los sensores más utilizados para la detección de obstáculos. En la sección III se describen los métodos en los cuales se obtiene información de profundidad. Por otra parte, en la sección IV se describen los métodos que no utilizan la información de profundidad. Para concluir, en la sección V se proporciona una vista general de los métodos descritos.

II. SENSORES PARA LA DETECCIÓN DE OBSTÁCULOS

Los sensores son dispositivos que evalúan información del entorno y producen una medición de este. Dependiendo de la información que proporcionan se pueden clasificar en exteroceptivos o propioceptivos. Los exteroceptivos proporcionan información del entorno, ejemplos de estos sensores son cámaras, escáneres láser, sonar, etc. Por otra parte, los sensores propioceptivos se encargan de obtener información de la misma plataforma como velocidad, aceleración y orientación. Uno de los sensores propioceptivos mas habituales son las unidades de medición inercial (IMU), las cuales están compuestas por acelerómetros y giróscopos. La detección de obstáculos es obtenida mediante sensores exteroceptivos, los cuales se encargan de recibir información del entorno. En algunos casos los algoritmos de detección de obstáculos necesitan una combinación de los dos tipos de sensores ya que a parte de necesitar información del entorno también necesitan información del movimiento de la plataforma.

Basándose en las tecnologías de los sensores, los exteroceptivos se pueden clasificar en activos o pasivos. Los sensores activos emiten una señal y procesan la reacción del entorno a esa señal. El uso de sensores activos para la detección de obstáculos es un procedimiento muy utilizado [20]. Por otra parte, existen los sensores pasivos, que no emiten ninguna señal sino que simplemente reciben información del entorno. Dos ejemplos habituales de esta categoría son las cámaras de espectro visible y las cámaras de espectro infrarrojo (IR). Las cámaras IR poseen la ventaja de poder detectar obstáculos como peatones durante la noche debido a la temperatura corporal [21], [22]. En [23] se presenta una comparación entre una cámara binocular IR y una de espectro visible para la detección de peatones.

A continuación se realiza una comparación entre los sensores activos y los pasivos. Dentro de los sensores activos, existen los llamados sensores de un solo haz; uno de los más importantes dentro de este grupo son los sensores de ultrasonidos (US). Éstos miden la distancia hasta el obstáculo más cercano en la orientación del sensor. Los sensores de US tienen la ventaja de tener un coste y peso reducido, además de resultar muy efectivos para algunas aplicaciones de detección de obstáculos [24]. Debido a la diferencia de información que proporcionan los sensores de US con los que se compararan a continuación estos no se deben tener en cuenta en la siguiente comparativa. Los sensores activos como Radar (del inglés radio detection and ranging) o Lidar (del inglés Laser Imaging Detection and Ranging) tienen un coste, peso y consumo de energía superior a los pasivos. Sin embargo, los sensores pasivos necesitan un procesamiento de la imagen mediante un ordenador para obtener la profundidad del entorno, mientras que en los sensores pasivos los datos crudos, es decir sin

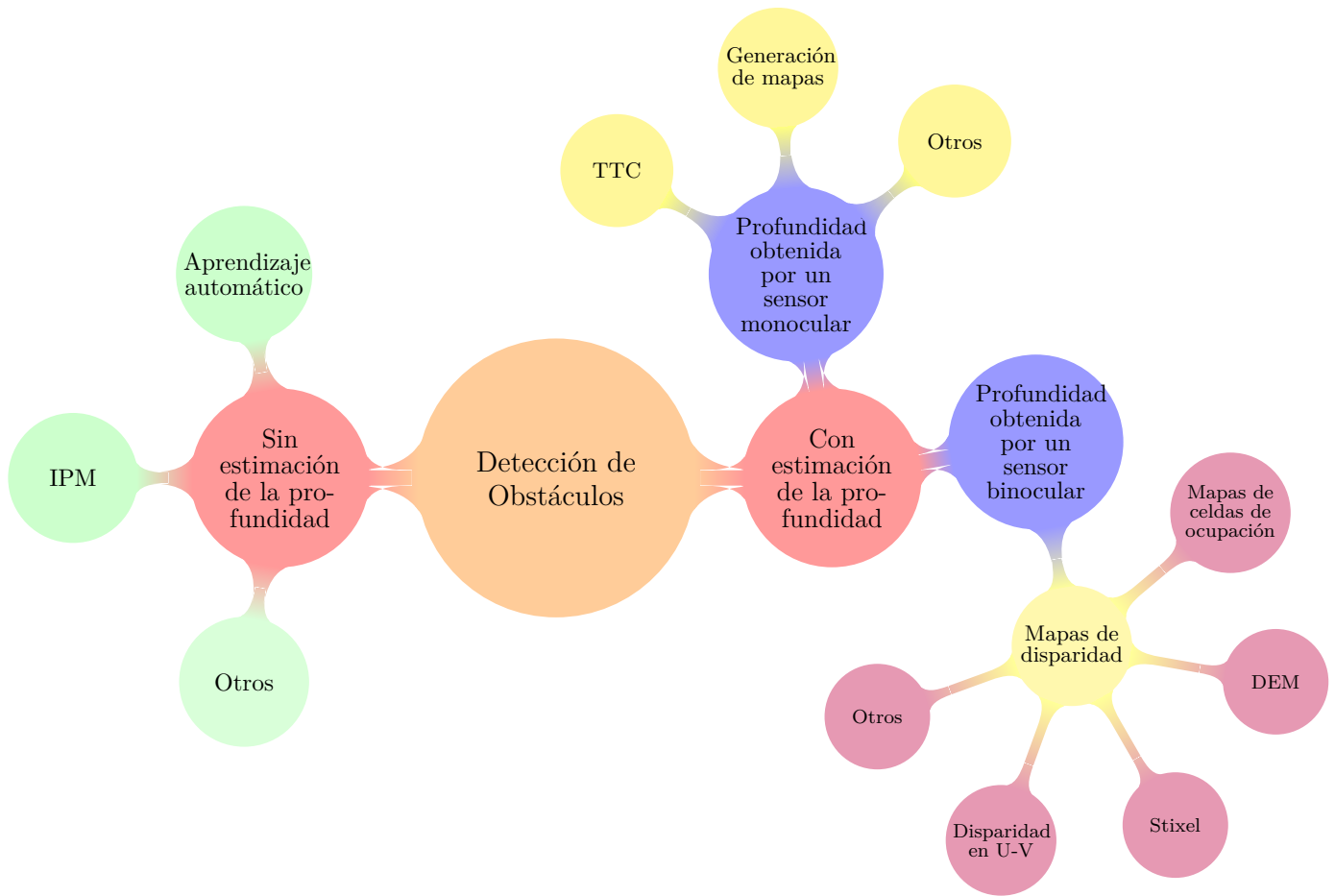


Figura 1: Taxonomía para clasificar los detectores de obstáculos basados en la visión según sus métodos.

procesar, ya proporcionan esta información. Además, en los sensores activos, la distancia a la que un obstáculo puede ser detectado es superior a la de los sensores pasivos. En entornos con mucha luz del sol los sensores activos pueden producir medidas erróneas, mientras que en entornos con poca luz éstos funcionan perfectamente, al contrario que los sensores pasivos. En la mayoría de algoritmos, los sensores pasivos necesitan de una calibración previa para evitar efectos como la distorsión, mientras que en los activos no es necesario. Los sensores activos necesitan emitir una señal, por lo que en lugares con múltiples sensores activos, estas emisiones pueden afectar a la medición produciendo medidas incorrectas.

Los sensores activos y pasivos tienen sus ventajas e inconvenientes, pero en muchos aspectos, las limitaciones de un tipo sensor se compensan con las del otro. Algunos métodos combinan ambos sensores para detectar obstáculos. Por ejemplo, en [25], los autores combinan una cámara monocular con un escáner láser. En [26] se utiliza un sistema binocular y un escáner láser.

Otro sensor activo, que utiliza un concepto diferente a los sensores de tecnología Radar y Lidar que no ha sido incorporado en la comparación y está ganando popularidad en la última década, es conocido como sensor RGB-D. Este sensor es capaz de obtener la información referente

a colores y profundidad de cada píxel de la imagen. Esta profundidad se obtiene mediante una cámara IR que se encarga de recibir y procesar un patrón IR emitido previamente por el mismo sensor. La información de color es obtenida mediante una cámara a color. Comparando este sensor con otros sensores pasivos, ambos tienen la ventaja de que no se ha de calcular la profundidad sino que el mismo sensor la proporciona. Por otra parte, el rango de detección de los sensores RGB-D suele tener un máximo de 6 metros, lo cual es una distancia inferior a los otros sensores activos, sin embargo, su peso es inferior pudiendo llegar a los 100 gramos.

En esta sección se ha presentado una vista general de los sensores utilizados para la detección de obstáculos junto con sus ventajas e inconvenientes. En [27] se presenta una explicación más detallada de los tipos de sensores existentes. Se ha de remarcar que en este estado del arte sólo los artículos que utilizan cámaras pasivas de espectro visible serán examinados.

III. MÉTODOS BASADOS EN ESTIMACIONES DE PROFUNDIDAD

La obtención de profundidad mediante sistemas visuales es una técnica muy utilizada para detectar obstáculos

del entorno. Esta profundidad proporciona una mejor definición de la posición del obstáculo, consiguiendo así poder utilizar algoritmos de navegación de más alto nivel. Es cierto que algunos algoritmos de esta sección son computacionalmente ligeros, aunque la mayoría de los métodos que generan mapas para poder detectar obstáculos implican una mayor utilización de memoria y procesador respecto a otros métodos. Otro problema que aparece en estos métodos con sensores visuales es que los valores de profundidad no pueden ser obtenidos o son inestables en entornos donde no hay características visuales.

En algunos de estos trabajos, la información de profundidad es almacenada a lo largo del tiempo, para poder realizar múltiples tareas entre las que se destaca la navegación de plataformas robóticas. Este almacenamiento de la información a lo largo del tiempo es conocida como mapeo del entorno [28]. Utilizando este enfoque, generalmente se necesitan manejar y almacenar grandes cantidades de datos. Si no se puede procesar toda la información se pueden generar retrasos, que pueden ocasionar que la detección del obstáculo se produzca más tarde de lo que se debería, y por consiguiente la plataforma puede colisionar con un obstáculo que no se ha detectado. Para evitar estos retrasos se aplican técnicas de reducción de datos. Un punto negativo de estas técnicas es que realizando esta simplificación se elimina información del entorno, pero dependiendo de la aplicación que se desarrolle, la simplificación no afecta al resultado. Un ejemplo de simplificación de datos es generar mapas que almacenen solo información relativa a los últimos instantes de tiempo. Estos mapas son conocidos como mapas locales. Por otra parte, los mapas que almacenan toda la información desde el momento inicial son conocidos como mapas globales.

Uno de los tipos de mapas que más se utilizan para la detección de obstáculos son los mapas de celdas de ocupación. Este tipo de mapa, a diferencia de los otros tipos, es explicado en este apartado, en lugar de en un subapartado como se hace con el resto, ya que éste es una representación utilizada tanto en sensores monoculares como binoculares. Los mapas de celdas de ocupación son una forma óptima de detectar obstáculos y reducir la gran cantidad de información que es obtenida por los sensores. Estos mapas presentan una solución simplificada de representación del entorno mediante una variable binaria que define si una celda está ocupada o libre. Aparte de binarizar el entorno, este tipo de generación de mapas agrupa la información proveniente del sensor en celdas, las cuales representan un volumen 3D o una área 2D del entorno. Con esta agrupación, se consigue reducir los numerosos puntos que ocupan una zona del entorno en un solo punto que representa una región ocupada o libre. El tamaño de la celda es predefinido antes de generar el mapa. Utilizando esta definición inicial existen una serie de problemas como que si hay un obstáculo muy grande y el tamaño de celda es muy pequeño se necesitan muchas celdas para representarlo, mientras que si el tamaño de celda es más grande que los objetos del entorno éstos ocuparán una región más grande de la que ocupan realmente. Definiendo un tamaño de celda más grande, el nivel computacional necesario, así como la precisión serán más bajos. Para poder este problema, en [29] se presenta un trabajo llamado Octomap.

Éste presenta un mapa de celdas de ocupación capaz de gestionar celdas de múltiples resoluciones basándose en octrees. Utilizando esta representación, es posible ajustar dinámicamente el tamaño de la celda en base al objeto a representar.

En algunos métodos, donde se requiere una generación de un mapa para poder detectar obstáculos, es necesario saber el movimiento de la plataforma robótica a lo largo del tiempo. Evaluando este movimiento a lo largo del tiempo, es posible conocer la posición del robot dentro del mapa. Esta localización se puede obtener mediante el procesamiento de imágenes (odometría visual), o mediante un sensor externo. El sensor externo propioceptivos más utilizado en entornos exteriores es el GPS, mientras que en entornos interiores se suele utilizar una IMU.

Existen varias técnicas para la obtención de la profundidad en sistemas visuales. Éstas son clasificadas según el sensor utilizado, monocular o binocular. En sistemas binoculares, dos imágenes son capturadas en un mismo instante de tiempo, detectando puntos característicos de cada imagen y analizando el desplazamiento de cada punto de una imagen con su correspondiente punto de la otra imagen. Sólo se puede calcular la profundidad de los píxeles que aparecen en ambas imágenes. Por otra parte, los sensores monoculares, al no disponer de dos imágenes en el mismo instante de tiempo, se tiene que realizar una evaluación de imágenes consecutivas. Una de las formas de obtención de la profundidad en sistemas monoculares es conociendo el desplazamiento de la plataforma entre las imágenes que se evalúan, posteriormente se realiza un casamiento de los puntos característicos de ambas imágenes. Uno de las mayores diferencias en la profundidad obtenida con ambos sensores es que utilizando una sola cámara no se obtiene una profundidad métrica sino que es una profundidad relativa a una escala, mientras que utilizando un sistema binocular se obtiene una distancia métrica. Otra diferencia es que los métodos utilizados en sensores binoculares son más ligeros computacionalmente ya que no necesitan estimar la posición de una cámara respecto a la otra, ya que las dos cámaras han sido previamente calibradas. Debido a la gran diferencia entre las formas de obtener profundidad entre los dos sensores, el siguiente apartado se distingue en base al sensor utilizado.

III-A. Profundidad obtenida por un sensor monocular

Existen múltiples formas de detectar obstáculos utilizando sistemas monoculares. La mayor parte de los métodos se basan en la evaluación de vectores de flujo y el análisis de los puntos característicos de una imagen. Estos puntos característicos, se utilizan para detectar cambios de escala o para la obtención de un mapa del entorno.

Entre los diferentes métodos de detección de obstáculos utilizados en sistemas monoculares caben destacar, los métodos de tiempo de contacto TTC y los métodos que utilizan generación de mapa. A continuación estos métodos son explicados en más detalle.

1) *Tiempo para el contacto o tiempo para la colisión (TTC)*: En el campo de la visión, el TTC es definido como el tiempo que un objeto del entorno tarda en colisionar o

alcanzar el plano de la cámara, considerando una velocidad constante. La técnica de TTC se basa en el mismo concepto que el utilizado en tareas de sistemas biológicos humanos tales como manipulaciones o desplazamientos por el entorno. En tareas de navegación de robots, el enfoque de TTC proporciona los reflejos básicos para evitar colisionar con obstáculos y conseguir realizar una navegación de forma segura. Uno de los métodos de navegación más utilizado en TTC se basa en la detección y el cálculo del tiempo para contactar con un obstáculo, una vez se obtiene este tiempo y se detecta que una colisión es inminente, se toma la decisión de girar o parar el movimiento.

Las ventajas de este enfoque es la demanda relativamente baja a nivel computacional, la capacidad de detectar obstáculos en un entorno sin conocimiento de la velocidad o la localización de la plataforma, ni la forma o el tamaño del obstáculo. Además, es capaz de detectar obstáculos dinámicos sin necesidad de una calibración previa de la cámara. Por otra parte, este enfoque tiene algunos inconvenientes como la necesidad de que la plataforma tenga movimiento, y, además, este movimiento se ha de corresponder con la dirección de la cámara. Este movimiento específico no siempre se corresponde en algunas plataformas holonómicas, tales como multirotores o algunas plataformas submarinas, mientras que en sistemas no holonómicos, como coches, robots terrestres no omnidireccionales, o plataformas de ala fija su desplazamiento es siempre en la dirección de la cámara. Es cierto que en ambos sistemas también se realizan movimientos de rotación, estos movimientos afectan a los cálculos del TTC. Sin embargo, en esta sección se citarán algunos métodos que consiguen eliminar estos efectos de rotación de la imagen.

Esta técnica puede proporcionar el resultado en profundidad, profundidad relativa, cantidad de imágenes o tiempo en segundos hasta la colisión.

Se pueden distinguir dos grandes grupos según la forma de calcular el TTC, los basados en flujo óptico y los que utilizan puntos característicos. Dentro del grupo de los métodos basados en flujo óptico, existe una técnica que se utiliza para simplificar la estimación del campo vectorial del OF. Ésta es conocida como foco de expansión (FOE) y es utilizada para la detección de obstáculos frontales. El FOE representa el punto en el plano imagen correspondiente con la intersección del vector velocidad. Describe el movimiento de la cámara y el plano de proyección. En otras palabras, el FOE es el punto donde hay menor magnitud y mayor divergencia de flujo. Obteniendo la relación entre la distancia del FOE al punto de interés en la imagen, y generalizando la magnitud del campo de flujo, se puede obtener un valor aproximado del TTC. Esta técnica es utilizada en [30] para detectar obstáculos para personas con discapacidad visual.

Otros enfoques descritos en [31],[32] utilizan el FOE para obtener mapas de profundidad para la detección de obstáculos. Este mapa de profundidad se obtiene mediante el cálculo del TTC para cada vector de flujo óptico. Posteriormente, éste es combinado con una estimación de la velocidad de traslación de la plataforma y la posición del FOE. Una de las suposiciones necesarias para la estimación del TTC es que el movimiento ha de ser lineal y frontal.

Si el movimiento no se produce en estas circunstancias, el FOE no se puede calcular correctamente sin una compensación de la rotación. Para compensar la rotación, algunos investigadores han utilizado el casamiento utilizando una ventana estándar (standard window matching), o algoritmos basados en el gradiente. Una de las carencias de estos algoritmos es la capacidad de solventar grandes desplazamientos. Una forma de resolver este problema es mediante un sistema de OF basado en múltiples escalas. Esta solución demuestra ser una forma práctica y fiable de realizar una segmentación de la imagen [33]. Otra forma de compensar el problema ocasionado por este movimiento angular es utilizando la información de la rotación proveniente de sensores propioceptivos, consiguiendo así tener solo información de traslación en la imagen [34].

Aparte del movimiento restringido de la cámara, otros factores que no permiten una buena estimación del FOE son el ruido en el cálculo del OF y la aparición de objetos dinámicos en la escena. Para obtener una buena estimación del TTC para la detección de obstáculos es necesaria que el FOE esté bien estimado. Para solventar este problema, en [35] los autores describen una estrategia utilizada en vehículos aéreos que permite filtrar el ruido de alta frecuencia que aparece en la imagen.

Otro enfoque para la obtención del TTC se basa en la selección y la extracción de puntos característicos. Estos puntos pueden ser bordes, contornos, esquinas, etc. Estos métodos producen una localización y una información más precisa de los puntos circundantes de la imagen. Estos métodos son descritos a continuación.

Los cambios de escala, mediante la evaluación de puntos característicos invariantes de escala relativa para la obtención de TTC, son utilizados también para la detección de obstáculos frontales. Estos puntos característicos tienen las ventajas de que son rápidos de calcular y son invariantes a los cambios de escala. En [36] los autores presentan una implementación que utiliza la transformación de puntos característicos invariantes a escala (SIFT) para detectar obstáculos mediante su cambio de escala. Los resultados que presentan son evaluados en un entorno simulado y el algoritmo no se ejecuta en tiempo real. En [37] se presenta un método capaz de detectar obstáculos frontales en un bosque mediante un algoritmo que se basa en la variación del tamaño de expansión. Éste utiliza Speeded-Up Robust Features (SURF) para la detección inicial de las regiones donde se pueden obtener los puntos característicos escalables, y posteriormente comparar el tamaño de la región alrededor del punto característico y obtener el TTC. Otro punto característico utilizado para estimar el TTC basándose en cambios de escala es el llamado Scale Invariant Ridge Segments (SIRS). En [38] se utiliza SIRS para hacer el seguimiento de múltiples obstáculos en tiempo real.

Una de las suposiciones del enfoque TTC es que el movimiento de la cámara respecto a la superficie u obstáculos ha de mantenerse constante respecto al tiempo. En [39] se presenta una evaluación de tres modelos de movimiento. Éstos se corresponden con velocidad constante, distancia constante y aceleración constante. Mediante una evaluación de los puntos característicos obtenidos mediante el

detector de esquinas de Harris y los cambios relativos de escala, los autores son capaces de seleccionar el modelo de movimiento apropiado para la estimación del TTC.

En [40], Alenyà et al. presentan un método basado en la extracción de puntos característicos de la imagen utilizando Active Contour Affine Scale (ACAS) para la detección de obstáculos utilizando TTC. Además, el método presentado es comparado con otros métodos que utilizan SIRS y la derivada de la intensidad de la imagen (IBD).

Otra forma de obtener TTC es descrita en [41]. Los autores proponen un método donde se obtienen los cambios de escala locales en cada punto característico obtenido por SIFT juntamente con una triangulación de Delaunay. Después de este proceso, se utiliza un filtro RANSAC para descartar valores atípicos y agrupar los puntos característicos en base a su movimiento afín. Sin embargo, este algoritmo es especialmente sensible a movimientos de rotación.

Una de las dificultades a la hora de obtener el TTC aparece cuando no existe textura en la imagen. Esto es debido a la ausencia de puntos característicos. En [42], se presenta un método que utiliza las líneas rectas de la imagen. Los autores parten de la suposición que muchos entornos con poca textura suelen ser ricos en líneas rectas. Utilizando la apariencia y las líneas del entorno, este método es capaz de detectar obstáculos cuando el OF no puede ser calculado, debido al ruido del OF o a la ausencia de textura. Otra solución al cálculo del TTC en entornos con falta de textura es propuesto en [43]. En este trabajo se propone una combinación de TTC y un análisis de la derivada de primer orden de la escena, utilizando una penalización en las regiones donde el OF es cero, debido a la ausencia de variación a causa de la falta de textura.

Hasta ahora, se han analizado métodos que utilizan el OF y los puntos característicos por separado para calcular el TTC. En [44], los autores presentan un algoritmo que combina ambos métodos para avisar a conductores de posibles colisiones frontales. Para ello, en la primera etapa se extraen puntos característicos utilizando el detector de esquinas de Harris, luego se obtiene el OF de los puntos característicos obtenidos previamente. Utilizando el vector de OF de los puntos destacados, el movimiento entre imágenes es obtenido mediante una implementación piramidal del rastreador de puntos característicos de Lucas Kanade. El TTC se obtiene mediante la combinación de la localización del FOE y la información del paso previo.

Existe otro método descrito en [45] que no utiliza ni OF ni puntos característicos para el cálculo del TTC. Este método se basa en un análisis del campo de movimiento resultante de un cuerpo rígido sobre una proyección de perspectiva asumiendo una intensidad constante. A diferencia de otros métodos de TTC, los métodos directos trabajan directamente con las derivadas de la intensidad de la imagen.

2) *Generación de mapas utilizando sistemas monoculares:* La detección de obstáculos mediante mapas es una técnica muy utilizada tanto en sensores monoculares como binoculares. Uno de los puntos importantes de esta metodología es que los obstáculos son localizados mediante

la evaluación del mapa generado. En esta sección, se presenta una vista general de cómo obtener la profundidad y posteriormente cómo incluirla en un mapa, los tipos de mapas que se utilizan y cómo se detecta un obstáculo en el mapa mediante sensores monoculares.

En la sección anterior se ha comentado la utilización de la estimación del FOE para la obtención del TTC para cada vector de flujo. En [46], los autores proponen un algoritmo capaz de obtener la profundidad sin conocer el FOE. En el primer paso, la profundidad es obtenida mediante una estimación de la profundidad asistida por posición (PADE), en donde es necesario el conocimiento de la rotación y traslación de la cámara. Este paso es obtenido mediante una técnica de OF. La profundidad calculada es almacenada a lo largo del tiempo en un mapa de celdas de ocupación. Además, en este trabajo, se compara el método propuesto con un método de TTC.

Otra forma de calcular la profundidad es utilizando el método de multi-view Stereo (MVS). Este enfoque es capaz de reconstruir un modelo o un entorno en tres dimensiones a través de una colección de imágenes. En [47], el MVS se utiliza para detectar obstáculos estáticos. En este trabajo se utilizan múltiples cámaras situadas en la parte frontal y en los laterales del vehículo. Para acumular toda la información que proporcionan las cámaras se utiliza un mapa de celdas de ocupación en dos dimensiones. Normalmente, en los mapas de celdas de ocupación, los obstáculos son detectados mediante la evaluación de las celdas que lo rodean, mientras que en este enfoque, los obstáculos son detectados utilizando el mismo principio que un láser. Utilizando este concepto, el mapa de dos dimensiones es segmentado utilizando una técnica que emula un sistema que emite un haz cada cierto ángulo. Estos haces definen la dirección y la distancia de la plataforma al obstáculo.

El cálculo de la profundidad también se puede obtener a través del seguimiento de imágenes utilizando diferentes puntos característicos. En [48], estos puntos característicos son extraídos utilizando el algoritmo de extracción de esquinas de Harris y son almacenados en un mapa disperso de tres dimensiones. Mediante la evaluación de este mapa se puede obtener una localización aproximada de los obstáculos. Otro trabajo en el cual también se utiliza el seguimiento entre imágenes de puntos característicos es presentado en [49]. La detección de obstáculos se obtiene mediante la evaluación de un mapa denso formado por una nube de puntos. Este mapa es proporcionado por un algoritmo de localización y generación de mapas de forma simultánea (SLAM).

Algunos algoritmos que utilizan nubes de puntos obtenidos por puntos característicos no se adaptan a algunos entornos muy específicos. Esto es debido a que algunos puntos característicos no son estables, son ocluidos o no son suficientemente discriminatorios para realizar un seguimiento. En el trabajo presentado en [50] se presenta una solución para detectar árboles con un cuatrirrotor en un entorno muy específico en el cual se presentan los problemas anteriormente comentados. Para ello se utiliza la geometría e información de muy bajo nivel para construir incrementalmente un mapa de los troncos más salientes. Para poder reducir la cantidad de datos mientras se ge-

nera el mapa, los autores presentan un enfoque inspirado biológicamente. Este enfoque detecta el movimiento de rasgos sobresalientes para solo introducir en el mapa los árboles cercanos al robot e ignorar los que están lejos. Para poder identificar estos árboles cercanos, las regiones de la imagen son divididas basándose en el paralelaje del movimiento.

3) *Otros*: En esta sección se presentan un grupo de trabajos que adaptan enfoques diferentes a los algoritmos basados en TTC o a los de generación de mapas para la detección de obstáculos.

La mayoría de los métodos de este estado del arte hacen referencia a la detección de obstáculos cercanos al sensor. Algunas técnicas como OF o las que obtienen profundidad mediante sensores binoculares suelen fallar cuando se quieren detectar obstáculos lejanos. En [51], se presenta un trabajo capaz de solventar este problema. El método propuesto se basa principalmente en dos etapas. En la primera, se generaran hipótesis mediante la estimación de la profundidad en el plano imagen de la escena. En la segunda, los candidatos obtenidos en la etapa anterior se evalúan mediante un análisis de la transformación de la perspectiva sobre el tiempo.

Otro grupo de trabajos utilizados en entornos muy específicos se concentran en realizar una navegación en entornos con pasillos, donde sólo es necesario detectar la distancia a los obstáculos laterales y ejecutar una tarea de navegación basándose en igualar la distancia entre los dos lados. Para obtener esta distancia lateral, en los trabajos [52], [53], [54], [55], se evalúa el OF obtenido por dos cámaras cada una localizada en cada lado de la plataforma. Este enfoque también puede ser abordado mediante la colocación de sensores activos tales como US a cada lado. Utilizando este enfoque solo se pueden detectar los obstáculos laterales mientras que existe el riesgo de colisionar con algún obstáculo frontal. En [56], los autores utilizan una cámara monocular con una lente de gran angular capaz de visualizar ambos tipos de obstáculos. Este método se basa en dos estrategias, la primera es capaz de detectar paredes laterales mediante el uso de la divergencia de vectores de OF sobre el FOE utilizando un modelo predefinido de divergencia. La otra estrategia se basa en la detección de obstáculos frontales mediante la búsqueda de valores atípicos sobre el mapa estimado.

Uno de los usos más comunes de las técnicas de aprendizaje automático es la detección de un obstáculo específico en lugar de identificar la profundidad o distancia a los obstáculos. Una forma de detectar obstáculos mediante estas técnicas es entrenando un modelo para que detecte obstáculos específicos. Basándose en esta suposición, cuando un nuevo obstáculo aparece en la imagen y éste no ha sido previamente introducido en el modelo, estos algoritmos tienden a fallar. Una solución a este problema es resuelta en los siguientes artículos mediante un entrenamiento del modelo basándose en obtener la profundidad de regiones de una imagen. En [57], se presenta un enfoque de aprendizaje supervisado, donde se muestra que son capaces de obtener la distancia en un coche de radio control que se mueve a gran velocidad en un entorno exterior no estructurado.

Otro trabajo similar capaz de aprender profundidad es presentado por Saxena et al. [58].

III-B. Profundidad obtenida por un sensor binocular

Un sistema binocular esta compuesto por dos cámaras desplazadas normalmente horizontalmente por una distancia conocida. El valor del desplazamiento recibe el nombre de baseline. Este desplazamiento se produce para poder observar el entorno desde dos puntos de vista diferentes en un mismo instante de tiempo. Comparando estas dos imágenes, se obtiene la profundidad de la escena, generalmente en forma de mapa de disparidad. A mayor distancia de baseline, mayor precisión en el calculo de la profundidad en los objetos lejanos, y menor precisión en los objetos cercanos, mientras que utilizando una menor distancia ocurre lo contrario. Para los puntos ocluidos entre imágenes no se puede calcular su profundidad.

Comparando con otras técnicas, los sistemas binoculares proporcionan una distancia métrica sobre los elementos del entorno. Esta distancia puede ser detectada en todos las direcciones de movimiento ya que la extracción de la profundidad no depende del instante anterior. Los sistemas binoculares son muy efectivos en helicópteros, multicopteros y vehículos submarinos operados remotamente (ROV) donde no existe movimiento cuando están en vuelo estacionario o se desplazan en cualquier dirección. Por otra parte, este sistema tiene desventajas, como por ejemplo, que el rango en el que los objetos son detectados esta acotado por el baseline. Otra desventaja es que la extracción de profundidad no se puede obtener en ciertas situaciones como en entornos uniformes, patrones repetitivos, entornos con poca luz, o en presencia de especularidades. También se ha de remarcar que estos sistemas necesitan de una calibración previa.

Tanto en sensores monoculares como binoculares, para una buena estimación de la profundidad es necesaria un proceso previo llamado rectificación de la imagen. El proceso de rectificación y el proceso de obtención de mapas de disparidad se pueden ejecutar en tiempo real en la mayoría de los ordenadores de a bordo de las plataformas robóticas. Cuando estas tareas son utilizadas por otros procesos como odometría visual, generación de mapas o navegación, la ejecución de todos ellos en tiempo real requiere de gran capacidad computacional. Es por esta razón que muchos autores ejecutan estos dos procesos en un hardware externo como Field Programmable Gate Arrays (FPGA), un ordenador con conjunto reducido de instrucciones RISC, siendo el caso más popular el de los procesadores ARM, o mediante procesadores gráficos (GPU). Con este hardware adicional se consigue aliviar computacionalmente el ordenador principal y, por consiguiente, éste puede ejecutar otras tareas de más alto nivel. En [59], se presenta una comparación entre las ventajas de utilizar FPGA o GPU para la rectificación y la obtención del mapa de disparidad. También se pueden ejecutar algoritmos de obtención de la profundidad y detección de obstáculos utilizando solamente una de estas placas de hardware adicional. Por ejemplo, en [60], los autores comparan la detección y evitación de obstáculos mediante un mapa disperso obtenido en un ARM y un mapa denso obtenido por una FPGA.

Los métodos de obtención de mapas de disparidad en sensores binoculares tienden a generar una gran cantidad de información. Por ejemplo, los métodos basados en Semi-Global Matching (SGM) y Sum of Absolute Differences (SAD) generan un mapa de disparidad denso del entorno, el cual si no es filtrado antes de la inserción en un mapa, puede provocar que el proceso deje de ejecutarse en tiempo real.

La gran mayoría de trabajos que se muestran en esta sección obtienen la información del entorno mediante mapas de disparidad. Por ello, se ha optado por una clasificación de los métodos de detección de obstáculos que utilizan sensores binoculares mediante el tipo de mapa utilizado para la detección de los obstáculos. Las mayores diferencias entre los diferentes métodos de representación son la precisión de la localización de los puntos del entorno y el grado de simplificación de la información proveniente del sensor hacia el mapa. Por otra parte, otra diferencia entre los métodos de representación es la necesidad de conocer o no la posición de la plataforma para poder generar un mapa incremental. Otros métodos no necesitan el conocimiento de la posición o el desplazamiento ya que sólo evalúan un instante de tiempo para realizar la detección de obstáculos.

Existen diferentes tipos de mapa que son utilizados para la detección de obstáculos, entre los que destacan los mapas de celdas de ocupación, los mapas digitales de elevación (DEM), los mapas representados por stixels o los mapas de disparidad U-V. En el ámbito de la robótica terrestre muchos enfoques se basan en la suposición de que el entorno es plano, mientras que otros realizan una segmentación del suelo respecto al resto de la imagen. A continuación se detallan todas estas técnicas.

1) Mapas de celdas de ocupación: Los mapas de cuadrícula de ocupación son una representación del entorno muy utilizada para simplificar la información que generan, entre otros, los sensores binoculares. Para generar este tipo de mapa es necesario conocer la posición de la plataforma en cada instante de tiempo.

En [61], los autores utilizan Octomap para representar el entorno y detectar obstáculos. La misma información binocular es utilizada para la generación del mapa y para la obtención de la posición de la plataforma robótica en cada instante mediante odometría visual. En este trabajo, se realizan varias etapas de simplificación de la información que proviene del sensor antes de introducirla en el mapa. Primero se reduce la resolución de la nube de puntos mediante un filtro, luego para generar Octomap se realiza otra reducción de puntos. Para concluir y solo centrarse en la información que consideran importante, todos aquellos puntos que no se corresponden con una capa horizontal del grosor correspondiente a la altura del cuatrirotor son suprimidos del mapa. Tal como se ha comentado anteriormente, este tipo de algoritmos necesitan gran capacidad computacional. Por ejemplo, en el trabajo de Schmid et al. [61] se utilizan varios componentes de hardware como una Intel Core 2 DUO CPU, una Gumstix y una FPGA. Además, la plataforma también requiere de un sistema binocular y una IMU.

Otra forma de poder reducir la cantidad de información proveniente del sensor es explicada en [62]. En este trabajo se utiliza un mapa de celdas de ocupación en tres dimensiones con un tamaño de celda previamente definido de 10 centímetros. Con este sistema, los autores son capaces de inspeccionar grandes superficies exteriores utilizando este detector de obstáculos en un helicóptero. En el trabajo de Heng et al. [63], para reducir la información proveniente del sensor utilizan una representación basada en un escáner virtual en tres dimensiones. Este escáner virtual se obtiene definiendo una esfera compuesta por varias cuadrículas de ocupación, las cuales representan espacios ocupados. Estos espacios ocupados son calculadas mediante la mediana de los rangos de los puntos que están en esa cuadrícula. Utilizando este proceso, la cantidad de información se reduce, así como la aparición de valores atípicos.

Hasta ahora se han visto técnicas que reducen la información antes de incorporarla en el mapa. En [64], se realiza la simplificación una vez se ha generado el mapa de cuadrículas de ocupación. Esta simplificación consiste en agrupar las celdas ocupadas en obstáculos individuales y luego cada obstáculo es convertido es un modelo poligonal de tres dimensiones.

Según la aplicación a realizar, sólo es necesario conocer la información relacionada con el entorno cercano. Una solución para este caso es la utilización de mapas locales en lugar de mapas globales. Los mapas locales permiten una rápida generación del mapa, y al mismo tiempo, un buen conocimiento del entorno circundante. Gohl et al. [65] presentan un trabajo en el que son capaces de detectar obstáculos en cualquier dirección utilizando una plataforma robótica aérea equipada con un conjunto de cuatro sensores binoculares y una IMU. Para la detección de obstáculos utilizan un mapa de celdas de ocupación 3D local. Para la de obstáculos se evalúa si la distancia de los puntos obtenidos en el mapa local por los cuatro sensores es inferior a un valor umbral. Para reducir la carga computacional del ordenador principal, los procesos de rectificación y cálculo de los mapas de disparidad son realizadas por una FPGA.

En muchas plataformas robóticas no es necesario el conocimiento de la información en tres dimensiones, ya que su desplazamiento solo se produce en dos dimensiones. Por ello, los puntos 3D se pueden proyectar a un mapa en 2D. Realizando esta simplificación, se reduce la cantidad de información y el detalle del entorno, pero al mismo tiempo, dependiendo de la aplicación, es una forma eficiente de reducir la cantidad de información de un mapa de cuadrículas de ocupación. Por ejemplo, en el trabajo de Lategahn et al. [66], se utiliza un mapa de cuadrículas de ocupación de dos dimensiones para detectar obstáculos en tiempo real en un coche.

Existen otros trabajos que utilizan el mapa de celdas de ocupación combinados con otros métodos para poder detectar obstáculos dinámicos.

En [67], se combina un mapa de cuadrículas de ocupación con un método de seguimiento para poder detectar obstáculos en el mar. Este método es utilizado para filtrar los falsos positivos ocasionados por la aparición de olas.

Sin embargo, en su trabajo anterior [68], se realiza la detección mediante un algoritmo basado en la detección de colores notorios y un análisis de movimiento disperso. Una de las desventajas de este método es que cuando la visibilidad es baja, la detección de colores notorios falla. Los obstáculos son detectados mediante la probabilidad de la medida a través de la evaluación de la altura, la cual almacena el valor más alto de los puntos proyectados en la celda. Posteriormente, estos valores son introducidos en un mapa de cuadrículas de ocupación de dos dimensiones. Otra forma de detectar obstáculos dinámicos es presentado en [69], donde los autores se ayudan de la odometría visual para distinguir entre objetos estacionarios y objetos dinámicos. Además de detectar ambos tipos de obstáculos, en este trabajo se calcula la velocidad de los obstáculos dinámicos y utiliza un filtro de Kalman para predecir la velocidad y reducir falsos positivos.

Los obstáculos dinámicos también se pueden detectar mediante una combinación de OF con mapas de disparidad. En [70], se utiliza el mapa de disparidad y los datos de flujo óptico para generar un mapas de cuadrículas de ocupación. Los obstáculos son representados en un mapa de celdas de ocupación, para posteriormente ser detectados y localizados. Los vectores de movimiento son obtenidos mediante un método de segmentación del suelo en planos. Los vectores que no encajan con la estimación son considerados obstáculos. Otro trabajo que utiliza los mismos métodos para la detección y localización de obstáculos es presentado en [71].

2) *Mapas digitales de elevación*: Los mapas digitales de elevación son una representación de la información del entorno que simplifica la información almacenada. La información utilizada para realizar esta representación suele ser una imagen de disparidad proveniente de un sensor binocular. Este enfoque se basa en una discretización de la altura, y representa las medidas del entorno en un mapa de celdas de ocupación cartesiano. El suelo se divide en cuadrículas, donde por cada celda se guarda la altura del punto más alto que ocupa cada cuadrícula y el resultado es una nube de puntos la cual contiene cada una de estas alturas para cada celda ocupada. Los DEM, al igual que los mapas de celdas de ocupación, son una representación binaria del entorno donde se representa si una región está libre u ocupada. Esta técnica además de utilizarse para la navegación y la detección de obstáculos, es también aplicada para el mapeo del terreno.

A diferencia de los mapas de celdas de ocupación, utilizando una agrupación de los objetos en bloques verticales se consigue reducir la aparición de falsos positivos y negativos. Además, con este sistema se consigue reducir la cantidad de puntos necesaria para representar el entorno ya que los puntos son agrupados en columnas de ocupación en lugar de cuadrículas.

Esta representación es comúnmente utilizada en vehículos terrestres. En [72], los autores son capaces de detectar y clasificar los obstáculos basándose en dos clasificadores. En el primero se evalúa la densidad de puntos por celda y con esta evaluación se es capaz de determinar si es carretera u obstáculo, mientras que en el segundo se utiliza un clasificador el cual es capaz de diferenciar

entre carretera, isletas o puntos que definen un obstáculo. Para detectar la superficie de la carretera se utiliza una combinación de RANSAC con un modelo de crecimiento de regiones. Utilizando el modelo de Oniga y Nedeveschi, los autores de [73] presentan un mapa digital de elevación basado en partículas. Este método parte de su antiguo trabajo [74], donde se creaba y se hacía el seguimiento sobre un mapa de celdas de ocupación. En el trabajo de Oniga y Nedeveschi, en cada celda del mapa, sólo se guarda la información de la altura, mientras que en el método de Danescu et al. también se guarda la distribución de probabilidad de la velocidad por cada celda, consiguiendo así describir los obstáculos dinámicos. En este trabajo, se asume que los objetos sólo se mueven en dos dimensiones. Por ello, el vector de velocidad sólo contiene dos componentes. En muchos casos, los sensores utilizados tienen limitaciones de precisión, rango y exactitud, las cuales producen incertidumbre. Por estas razones y para poder minimizar estos problemas, los autores proponen un método de monitorización de las medidas y calculan una densidad de probabilidad para cada celda en el mapa.

Otro trabajo que también utiliza un método de monitorización, en este caso ayudado por la odometría, es presentado en [75]. Los autores describen un enfoque que segmenta la información de entrada en diferentes capas. Uno de los problemas de este método es que es difícil evaluar la consistencia de un obstáculo cuando no se tiene la información de una capa. Por eso, los autores recurren a la utilización de un método de monitorización.

3) *Stixel*: Otra representación del entorno en forma de mapa que reduce la cantidad de información es conocida como stixel. Este sistema proporciona una representación de nivel medio del entorno utilizando un conjunto de barras verticales. Cada una es definida por su posición relativa a la cámara y en posición vertical desde el suelo hasta una altura determinada, y representa la posición de obstáculos en el entorno. La generación de mapas utilizando esta técnica simplifica el entorno mediante una representación basada en superficies horizontales y verticales. Las superficies horizontales representan carreteras y zonas transitables, mientras que las superficies verticales representan obstáculos como peatones, coches, semáforos, etc. Al igual que los mapas de elevación digital, la representación con stixel es comúnmente utilizada en vehículos terrestres.

El trabajo inicial para la detección de obstáculos mediante este enfoque es abordado en [76]. El proceso de la creación de esta representación empieza con un mapa de ocupación polar obtenido mediante la información proveniente de un sensor binocular y se calcula la altura del suelo basándose en el espacio libre de la etapa anterior. Por otra parte, la altura de los stixels se obtiene mediante una segmentación de las imágenes de disparidad en primer plano y fondo. Aunque éste sea el método más común, existen variaciones como en el caso [77] donde no se utilizan los mapas de disparidad, y por consiguiente, el coste computacional es inferior.

Una de las ventajas respecto a otros métodos de segmentación de carreteras o caminos es que la representación en stixels no se basa en una segmentación de la carretera completamente plana sino que usan una representación

de ésta basada en una superficie B-spline [78]. Por otra parte, uno de los problemas que afectan directamente a la generación de mapas en general es la calidad de los datos de disparidad. Este mal cálculo de la disparidad puede ser provocado por múltiples factores. Por ejemplo en [79] el mapa utilizado contiene errores de disparidad ya que es utilizado en entornos con malas condiciones meteorológicas. Para solventar este problema, a diferencia de su antiguo trabajo [80], los autores presentan una solución para la representación basada en stixels la cual utiliza una formulación Bayesiana y métodos de estimación existentes.

En [81], se muestra una variación del algoritmo original de stixel en el cual se obtiene una mejor precisión en la segmentación. Esto se consigue mediante una información de enfoque descendiente en lugar de la propuesta descrita anteriormente. La segmentación es modelada utilizando los campos aleatorios de Markov y solucionado mediante programación dinámica. Este método es evaluado mediante la utilización de un coche en una autopista y es capaz de detectar la posición de los coches y los guardarailes en la imagen, así como la detección de regiones libres de obstáculos.

Es bien sabido que la mayoría de entornos relacionados con coches incluyen obstáculos dinámicos tales como coches o peatones. Para una navegación segura de la plataforma en estos entornos es necesario poder distinguir entre objetos estacionarios y dinámicos. Uno de los métodos más utilizados para realizar esta distinción son los métodos de seguimiento. En [82], se basan en este concepto de seguimiento para identificar ambos obstáculos mediante la representación basada en stixels. El movimiento tanto lateral como longitudinal de cada stixel es estimado usando un filtro de Kalman. En [83], los obstáculos dinámicos son obtenidos mediante una segmentación de la escena en múltiples clases de tráfico, basándose en el trabajo [82], el cual demuestra una representación del objeto basada en super-píxeles. En el trabajo de Erbs et al. [83] incorporan la velocidad de cada stixel además de su posición y su altura. Respecto a la velocidad, se asume que los objetos se mueven en un suelo plano. Por eso, ésta se realiza en dos dimensiones. Otro trabajo en el cual se es capaz de detectar ambos tipos de obstáculos es presentado en [80]. Éste se basa en el trabajo de Erbs et al.

4) *U-V disparidad*: La disparidad en V es un método de bajo coste computacional capaz de detectar obstáculos utilizando como entrada al sistema la información proveniente de un mapa de disparidad. La V significa la localización del píxel en coordenadas de la imagen (u, v). La disparidad en V simplifica la extracción de obstáculos de tres a dos dimensiones mediante un proceso lineal. Acumulando los píxeles con la misma disparidad en cada fila, los obstáculos perpendiculares pueden ser representados mediante píxeles verticales cuya intensidad representa la anchura de los obstáculos.

Este método es una solución óptima para procesar información a una frecuencia alta en ordenadores con poca capacidad computacional. Esto es debido al bajo cómputo necesario para convertir imágenes de disparidad a un mapa representado por la disparidad en V. En [84]

los autores comparan este enfoque con un mapa de celdas de ocupación [61].

La disparidad en V fue originalmente utilizada para detectar obstáculos tanto en carreteras planas como no planas [85]. En [86], se presenta la disparidad en U. Este enfoque es muy similar al de la disparidad en V, pero en lugar de acumular los píxeles con la misma disparidad para cada fila, en éste se acumulan respecto a las columnas, obteniendo así la altura de cada obstáculo. En algunos trabajos, estos mapas de disparidad son utilizados conjuntamente, creando así un método conocido como disparidad en U-V. Este tipo de mapa permite obtener tanto la altura como la anchura de los obstáculos.

Este enfoque es muy utilizado para segmentar el entorno en superficies que corresponden a carretera y superficies que no son carreteras. Aunque en algunos de los siguientes trabajos los autores no se centran específicamente en detectar obstáculos en un entorno, éstos detectan regiones del entorno transitables. Una de las ventajas de los mapas de disparidad en U o V para la detección de zonas transitables en carreteras es que no es necesario obtener puntos característicos, ni extraer bordes de la carretera, ni asumir una carretera plana. Además, otra utilidad importante de la disparidad en U-V es la capacidad de obtener información referente a la forma y posición del suelo.

En [84], se presenta un trabajo en el cual se realiza una segmentación del suelo mediante el análisis del mapa de disparidad en V. Éste se extrae utilizando la diagonal que se genera cuando se proyecta el plano del suelo. Sin embargo, los obstáculos son obtenidos directamente de la representación de disparidad en U. Existe otro trabajo [87] que en lugar de utilizar el mapa de disparidad en V, como el trabajo de Oleynikova et al., se determina el espacio libre de obstáculos mediante dos etapas. En la primera se obtiene el mapa de disparidad en U, mientras que en la segunda etapa se realiza una segmentación basada en color para extraer la carretera.

Otra forma de identificar la línea que separa el suelo del resto, por el caso de los mapas de disparidad en V, es extrayéndola mediante la transformada de Hough [88], [85], [86], [89]. Sin embargo, esta forma de extracción puede ser incorrecta o ineficiente cuando las líneas candidatas obtenidas por Hough no han sido correctamente clasificadas. Para solventar este problema, en [90] se propone un algoritmo de detección de regiones transitables, donde se introduce el concepto de disparidad o no disparidad del obstáculo. Con este concepto, las líneas que aparecen en las regiones no transitables son reducidas. Utilizando este método, las regiones transitables pueden detectarse incluso cuando sólo aparecen en una pequeña parte de la imagen.

Hasta ahora, los mapas de disparidad en U-V han sido utilizados de forma independiente. Sin embargo, existen una serie de trabajos donde éstos son utilizados conjuntamente para poder diferenciar los obstáculos del suelo [91], o bien clasificar la carretera en superficies relativamente planas y caracterizar los puntos característicos de superficies del pavimento de la carretera, estructuras de los laterales de la carretera y obstáculos [92].

Utilizando los mapas de disparidad UV combinados con métodos de seguimiento se pueden detectar obstáculos dinámicos. En [93], se presenta un trabajo donde se detecta el movimiento no rígido de peatones como un sistema de ayuda a la conducción de automóviles. Para ello, se realiza una extracción de segmentos en el mapa de disparidad UV consiguiendo obtener los límites del obstáculo. Una vez el obstáculo es detectado, éste es seguido mediante un proceso de rastreo basado en un modelo.

En [94], los autores combinan el mapa de disparidad en V para extraer la información relativa a la posición y la forma del suelo. Luego los obstáculos son detectados mediante el espacio de disparidad de la imagen (DSI).

5) *Otros*: Existen un conjunto de enfoques los cuales no utilizan la generación de mapas para la detección de obstáculos. Éstos tienen la ventaja de que, al no utilizar un mapa, el coste computacional es inferior respecto a los que generan mapas.

En algunas plataformas robóticas aéreas no se pueden incorporar placas de procesamiento muy pesadas debido a su diseño y a su aplicación. Por esto, se necesita que los algoritmos utilizados requieran de bajo poder computacional para ser implementados en electrónicas simples. Por ejemplo, en [104] se presenta un flapping wing de 21 gramos que incluye unos sensores y una electrónica de procesamiento de datos muy sencillos. Con este sistema, los autores son capaces de detectar la distancia a un obstáculo mediante la evaluación de la imagen de disparidad.

Otro ejemplo en el que no se generan mapas para la detección de obstáculos es presentado en [105]. Los obstáculos son detectados directamente en la imagen en lugar de extraer el mapa de disparidad. Con este método se pueden detectar tanto obstáculos móviles como estáticos a largas distancias. Este método se basa en pruebas de hipótesis estadísticas usando un criterio geométrico local. Sin embargo este algoritmo solo funciona con superficies planas.

Si se compara con otras técnicas, la utilización de los mapas de disparidad presenta una gran serie de ventajas. Por otra parte, ésta tiene algunas carencias que pueden ser solventadas con la combinación de estos mapas con otras técnicas tales como OF, aprendizaje automático, etc.

Una técnica que utiliza un sensor binocular y que no usa la generación de mapas para la detección de obstáculos es conocida como el flujo binocular o flujo de la escena. Éste es una combinación de dos técnicas muy conocidas, el OF y la disparidad binocular. Una peculiaridad del flujo binocular es que la mayoría de métodos que utilizan esta técnica son capaces de detectar objetos móviles. Utilizando esta técnica, la segmentación que se realiza con el OF y la disparidad binocular es más segura y el reconocimiento de los obstáculos móviles es más rápido. Una de las razones de porqué este reconocimiento es obtenido más rápidamente que otros métodos es debido a la capacidad de analizar obstáculos con cierta dirección y movimiento. Esta técnica se utiliza principalmente para la detección de peatones o coches en movimiento.

A continuación se analizan algunos métodos que uti-

lizan el flujo de la escena. Una de las formas de detectar obstáculos en movimiento es separándolos del fondo estático. Este proceso se dificulta cuando el sensor encargado de analizar el entorno está en continuo movimiento. Por esta razón, el movimiento propio del vehículo ha de ser estimado. En [106], los autores comprueban cada vector de movimiento obtenido por el flujo óptico con el mapa de disparidad para detectar obstáculos dinámicos. Un trabajo similar es descrito en [107] donde se detecta y se segmentan los obstáculos en situaciones muy dinámicas utilizando el seguimiento de puntos característicos dispersos sobre múltiples instantes de tiempo. Éste detecta los obstáculos mediante la estimación y el desplazamiento en tres dimensiones de puntos seleccionados. En [102], se presenta un trabajo similar al de Vaudrey et al. [107], donde la detección se realiza utilizando campos de flujo de la escena densos evaluados en dos instantes consecutivos. Este algoritmo estima la probabilidad del movimiento para cada píxel de la imagen, y, basándose en su movimiento en tres dimensiones, evalúa si es una región con objetos que se mueven de forma independiente.

En [103], los autores proponen una combinación de un algoritmo de flujo disperso de la escena combinado con un seguimiento de los objetos sobre múltiples fotogramas. Los puntos adyacentes de la escena con flujo similar son considerados como un objeto rígido.

Una de las dificultades que presenta la detección de obstáculos mediante sensores binoculares es que los obstáculos muy lejanos son difíciles de detectar ya que éstos son muy pequeños y el ruido ocasionado por el cálculo de la profundidad no permite una correcta segmentación entre el obstáculo y el fondo. Para solventar este problema, en [108] se propone un método que combina múltiples señales para realizar la segmentación de objetos, una disparidad de alta precisión y una estimación del flujo óptico. Utilizando esta combinación de métodos, los autores muestran una segmentación capaz de distinguir entre vehículos y fondo en un alcance de hasta 180 metros.

Otro método que se centra en la mejora de la precisión de la disparidad es presentado en [109]. Los autores son capaces de segmentar y estimar la posición y el movimiento de obstáculos dinámicos mediante una combinación de un sensor binocular y la evaluación del movimiento. La clasificación entre los dos tipos de obstáculos, los estáticos y dinámicos, es realizada utilizando un filtro de Kalman para realizar el seguimiento de puntos en tres dimensiones, y compensar el propio movimiento de la plataforma. Por otra parte, otro método muy similar y con los mismos objetivos que el trabajo de Rabe et al. [109] es presentado en [110]. Los obstáculos son detectados mediante la estimación de la profundidad y del movimiento utilizando filtros de Kalman. A diferencia del método previo, la segmentación de los obstáculos no es necesaria.

Otra forma de detectar obstáculos dinámicos es utilizando de forma independiente un algoritmo de OF y el mapa de disparidad. En [70], los obstáculos son detectados y representados en un mapa de ocupación. A diferencia de otros métodos, en este trabajo el mapa de disparidad y los datos de OF son introducidos cada uno por separado en un mapa de cuadrículas de ocupación. Los vectores de

Cita	I/O	Loc Req	Tipo de vehículo	S/D	Mapa	Casamiento	Procesador	Freq
[65]	O	IMU	MAV	D	3D	SGM	FPGA + CORE I7	80
[60]	I/O	No	Ala fija	S/D	No	SGM	ARM VS FPGA	120
[95]	I	NO	MAV	D	3D	SGM	FPGA + ASCTEC I7	15
[96]	O	IMU	Coche	D	No	SGM	Core 2 Quad	10
[97]	I/O	IMU + VO	UGV	D	2.5D	SGM	FPGA + Core I7	14.6
[61]	I/O	IMU + VO	MAV	D	3D	SGM	FPGA + C2D + Gumstix	14.6
[98]	O	No	Coche	D	–	ELAS	–	–
[62]	O	IMU + GPS	UAV	D	3D	SAD	Pentium 1.6 GHz	25
[99]	O	–	Avión	D	–	SAD	–	–
[100]	O	No	Tractor	D	3D	SAD	–	–
[101]	O	–	Aéreo y terrestre	D	–	SAD	–	–
[102]	O	No	Coche	D	–	SGM	–	–
[103]	O	No	Coche	S	–	ELAS	C2D	–
[63]	I	IMU + GPS	MAV	D	3D	–	C2D	10
[67]	O	GPS	Barco	D	3D	SSD	–	–
[75]	O	No	Vehículo terrestre	–	3D	–	Core I7	–
[84]	O	No	MAV	D	2D	SGM	FPGA + CPU	60
[64]	O	No	Helicóptero	–	2D	–	Core 2.15 GHz	–
[104]	I	No	Flapping Wing	S	–	Mod. SAD	ATM328P 20MHz	15-30
[105]	O	No	Coche	–	–	–	NVIDIA GeForce	20
[69]	O	VO	Coche	D	3D	Mod. SGM	Core I7	10

– significa que la información de esta columna no esta disponible.

I/O: se indica si un método se utiliza en un entorno interior (I) o exterior (O).

Loc Req.: se refiere si la localización es necesaria en el proceso de detección de obstáculos. VO se refiere a odometria visual.

S/D: se indica si el método es disperso (S) o denso (D).

Casamiento: en esta columna se citan los métodos utilizados para obtener los mapas de disparidad. Los siguientes acrónimos vienen del inglés, SGM significa semi-global matching, ELAS significa efficient large-scale stereo matching, SAD significa sum of absolute intensity differences, SSD significa disparity estimation for dynamic stereo, mientras que mod. indica que se han hecho modificaciones respecto al algoritmo original.

Procesador: En esta columna se muestran los procesadores utilizados, C2D se refiere a Core 2 Duo.

Freq: en esta columna se muestra la frecuencia a la que el algoritmo es ejecutado en Hz.

Tabla I: Vista general de las principales métodos de obtención de profundidad

movimiento son obtenidos mediante un método de segmentación de suelos planos. Entonces, los vectores de flujo óptico que no encajan con la estimación son considerados como obstáculos. Otro trabajo que utiliza un concepto similar para la detección y localización de obstáculos es presentado en [71].

En [111], el flujo óptico y el resultado de la extracción de puntos densos de un sistema binocular son utilizados para representar el entorno así como obtener una estimación del movimiento propio de la plataforma. Combinando esta información son capaces de detectar personas y vehículos en movimiento.

Hasta ahora, los trabajos que se han citado combinan la información del sistema binocular con la información del flujo óptico para detectar obstáculos en una misma región. Sin embargo, en [101] y [56] se presentan dos métodos que no se combinan sino que cada uno sirve para detectar obstáculos dinámicos en diferentes áreas de la plataforma, en este caso, laterales y frontales. Ambos trabajos utilizan el flujo óptico para detectar obstáculos laterales, mientras que la principal diferencia es la detección de obstáculos frontales. En el trabajo de Hrabar et al. [101], se utiliza un sistema binocular para obtener imágenes de disparidad y detectar obstáculos mediante ellas, mientras que en Stowers et al. [56], se utiliza un sistema monocular. El número de sensores para la detección de obstáculos varía entre los dos trabajos, en uno solo se utiliza una cámara monocular con un gran campo de visión para detectar obstáculos laterales y frontales [56], mientras que en el otro [101] se utilizan múltiples cámaras para el mismo propósito.

Existen una gran variedad de métodos que combinan flujo óptico con la información de disparidad obtenida por un sensor binocular. Aparte de esta combinación, la disparidad se puede combinar con otros métodos. Por ejemplo, en [112], se presenta una combinación de técnicas que utilizan información semántica de los obstáculos para categorizarlos y realizar un seguimiento de ellos. Este trabajo se utiliza para la detección y categorización de coches, peatones y obstáculos estáticos. La detección de coches y peatones se realiza mediante una combinación de la evaluación de histogramas de gradientes orientados (HOG) y una máquina de vectores de soporte (SVM). Mientras que los obstáculos dinámicos son detectados y almacenados en un versión modificada de un mapa de celdas de ocupación [113]. Otro trabajo basado en información semántica es propuesto en [114]. Este enfoque es capaz de detectar la parte trasera de los coches mediante un operador de simetría e información en tres dimensiones de una cámara binocular. A diferencia de su trabajo anterior, Toulminet et al. [114] utilizan la información de un sistema binocular para extraer contornos en tres dimensiones de los obstáculos. Posteriormente, combinan un operador de simetría con el casamiento de un modelo simplificado de la parte posterior de un vehículo y la información proveniente de un sistema monocular.

Otro trabajo que utiliza una combinación de la información de disparidad es presentado en [115]. En éste información de disparidad es utilizada para entrenar un sistema de aprendizaje on-line. El clasificador es utilizado para detectar obstáculos a gran distancia. Este sistema utiliza una nube de puntos de la cual se realiza una segmentación entre puntos clasificándolos en obstáculos, footline

y tipos de suelo. Cuando se dispone de la información de las etiquetas, el sistema de aprendizaje on-line obtiene los puntos característicos visuales y éstos son utilizados para entrenar el clasificador.

La combinación de flujo óptico con información de profundidad permite la segmentación y detección de objetos lejanos. En [68], se describe un método para un entorno muy específico, el cual utiliza una segmentación de la imagen utilizando una técnica monocular, mientras que, la profundidad de los objetos segmentados es obtenida mediante la información binocular. Para realizar esta segmentación primero se detecta la línea que separa el cielo del mar utilizando un análisis de los píxeles mediante RANSAC. A continuación, las regiones detectadas como obstáculos son marcadas y, posteriormente, se obtiene su profundidad mediante información binocular.

IV. MÉTODOS QUE NO UTILIZAN ESTIMACIONES DE PROFUNDIDAD

Una de las aplicaciones principales de la detección de obstáculos es poder encontrar un camino donde poder navegar de forma segura, esto es, sin colisiones con los obstáculos. Para realizar esta navegación de forma segura, no siempre es necesario disponer de la información de profundidad del entorno. En muchos casos sólo es necesario poder identificar en qué parte de la imagen se detectan obstáculos y qué regiones están libres; es decir, diferenciar entre regiones transitables y no transitables. Esta clasificación se puede obtener mediante diferentes métodos, entre los que destacan los de segmentación de la imagen, los que evalúan el efecto de las regiones que se acercan (looming), o mediante la evaluación de la diferencia de la posición de los píxeles entre dos imágenes conociendo el movimiento propio de la plataforma y eliminando el efecto de perspectiva de la escena.

IV-A. Aprendizaje automático

Una de las ventajas del aprendizaje automático es la capacidad de aprender una habilidad sin ser explícitamente programada. Dentro de este ámbito, existe un grupo de métodos los cuales detectan obstáculos basándose en la segmentación de las imágenes. Realizando una clasificación de cada región es posible identificar obstáculos o espacios libres en la imagen.

En [116], los autores presentan un enfoque de aprendizaje automático capaz de detectar obstáculos sobre una plataforma aérea utilizando un campo aleatorio de Markov (MRF). Este trabajo modela los obstáculos como una función referente a puntos característicos visuales y dependencias no locales en las regiones vecinas de la imagen. Otro trabajo que utiliza MRF [117] sirve para detectar obstáculos en un vehículo de superficie no tripulado (USF). Este enfoque no necesita extraer puntos característicos de la imagen, sino que el modelo determina que la escena ha de ser descompuesta en tres componentes dominantes las cuales se diferencian mediante propiedades semánticas y visuales.

Otra forma de detección de obstáculos mediante segmentación es presentado en [118] y es utilizado para la

detección de obstáculos en una plataforma aérea. Este trabajo utiliza un método de aprendizaje automático para detectar y segmentar el cielo en la imagen. Una vez segmentado el cielo del resto, todos los objetos que aparezcan en la región de cielo son identificados como obstáculos. Uno de los inconvenientes de este método es que sólo es capaz de detectar obstáculos sobre la línea de horizonte. Esto quiere decir que a una altura inferior a esta línea no es capaz de detectar obstáculos.

Un trabajo con una detección de obstáculos muy específico es detallado en [119]. Los autores son capaces de detectar coches durante la noche mediante una evaluación en blanco y negro de las luces de estos. Primeramente, las luces son identificadas del resto del entorno mediante un análisis de todos los píxeles de la imagen. Si su brillo supera un valor umbral, éstos son considerados luces. Una SVM es utilizada para identificar entre las luces provenientes de vehículos y otros tipos de luces.

En lugar de detectar obstáculos en el entorno, existen métodos que utilizan una detección de lugares transitables o regiones libres de obstáculos. Por ejemplo, en [120] se utiliza una técnica de aprendizaje automático capaz de detectar espacios libres en entornos submarinos. Primeramente los autores definen un lugar donde se observa agua. Utilizando esta información, en la etapa de entrenamiento se aprenden las características de los superpíxeles donde hay agua. Cuando el proceso de entrenamiento ha terminado, un algoritmo es capaz de clasificar los superpíxeles de cada imagen como una región donde hay o no hay agua.

Tal como hemos visto anteriormente, muchos de los vehículos terrestres son capaces de detectar obstáculos como paredes, aceras, peatones, coches, etc. Sin embargo, dependiendo del entorno, algunos obstáculos pueden presentarse en forma de irregularidades o complejidad del terreno. En [121] se presenta un método capaz de detectar estas irregularidades. Éste utiliza un método de aprendizaje on-line basado en aprendizaje autónomo. Por otra parte, en [122] las regiones de la imagen que pueden ser transitables son detectadas utilizando una combinación de flujo óptico inverso y un aprendizaje autosupervisado.

Algunos métodos son capaces de detectar obstáculos mediante un enfoque inspirado biológicamente llamado Lobula Giant Movement Detector (LGMD). El LGMD es una neurona que está localizada en el cerebro de las langostas. Esta neurona se estimula cuando se aproxima un objeto en rumbo de colisión directa. Este modelo se suele utilizar en plataformas robóticas donde se navega en entornos impredecibles y no se utilizan algoritmos de reconocimiento de objetos.

En [123], se puede encontrar una evaluación de este método sobre un robot móvil. Un enfoque muy similar al de Blanchard et al. [123] es utilizado en [124] para detectar colisiones en automóviles. Otro trabajo que utiliza este enfoque en automóviles es presentado en [125]. Este último combina LGMD con un estimador de características topológicas y un algoritmo de enfoque de la atención. Además, para filtrar falsas alarmas, se evalúa la forma de los objetos para identificar personas, líneas de la carretera o coches.

Uno de los puntos débiles de LGMD es la poca capacidad de distinción entre objetos que van a colisionar y los movimientos de rotación de la plataforma. Stafford et al. [126] presentan un enfoque basado en la combinación de LGMD y un detector de movimientos elementales (EMD), el cual es capaz de detectar movimientos en direcciones específicas.

Para concluir con esta sección, en [127] se presenta una solución basada en LGMD la cual esta implementada en una FPGA. Tal como se ha podido observar, LGMD es principalmente utilizado en automóviles. Sin embargo, este enfoque también se ha implementado en otras plataformas, tales como robots aéreos [128], [129].

IV-B. Mapeo de perspectiva inversa

El concepto de mapeo de perspectiva inversa (IPM) se basa en la obtención de la transformación geométrica que permite eliminar los efectos de la perspectiva de la escena, reasignando en el dominio de las dos dimensiones en la cual el contenido de la información se distribuye homogéneamente sobre todos los píxeles. IPM se basa en la idea de invertir el mapa de perspectiva de los puntos en tres dimensiones a un plano imagen donde se supone que todos los puntos pertenecen a un plano conocido. Para obtener la información de la transformación es necesario conocer algunas condiciones de adquisición como la posición, la orientación y la óptica de la cámara, así como algunas suposiciones de la escena que se representa en la imagen. Este método fue presentado inicialmente por [130], mientras que en [131] fue posteriormente adaptado para la localización de obstáculos.

Utilizando esta técnica, los obstáculos son detectados mediante una comparación de puntos característicos de dos imágenes. Aplicando el movimiento entre las dos imágenes, sus puntos han de coincidir. En el caso de que los puntos no coincidan, estos son clasificados como puntos elevados y consecuentemente como un obstáculo. Para poder aplicar el movimiento entre las dos imágenes, es necesario saber el movimiento propio de la plataforma. En [132], [133] y [134] Bonin et al. utilizan IPM para la detección de obstáculos en una plataforma terrestre.

En ocasiones, el movimiento o vibraciones de la plataforma puede provocar la inestabilidad en las imágenes. Esta inestabilidad puede provocar una mala estimación del movimiento de la plataforma. Nieto et al. [135] presentan un algoritmo capaz de estabilizar imágenes mediante el uso de la estimación del punto de fuga, lo cual ayuda a las tareas de detección y seguimiento. Éste es evaluado en un coche para detectar obstáculos tanto en autopistas como en carreteras locales. Otro trabajo que se centra en detectar movimientos que pueden provocar la inestabilidad en la obtención de las imágenes es presentado en [136]. En éste se realiza una evaluación de la imagen entera para extraer y posteriormente eliminar los efectos producidos por el movimiento de cabeceo del vehículo. Este método es utilizado previamente a la detección de regiones libres delante de un coche en autopistas.

En el artículo [137], los autores no necesitan conocer el movimiento del propio vehículo ya que utilizan una eficien-

te estrategia jerárquica. Este trabajo es capaz de detectar tanto objetos dinámicos como estáticos. IPM es utilizado como un primer paso para la detección de zonas donde hay obstáculos. Un algoritmo de segmentación es el encargado de realizar una detección más precisa. Este algoritmo se utiliza para la detección de obstáculos traseros de un coche mediante la utilización de una cámara monocular.

Aunque el IPM no sea computacionalmente pesado, comparándolo con los métodos que generan mapas, existen algunos trabajos [138], [139] que han conseguido reducir el tiempo computacional. Esta reducción se obtiene mediante la implementación de un perfil vertical de una dimensión en la detección del IPM aplicado en una región de interés. Este sistema es conocido como franja de detección de peatones (PDS). Los autores utilizan PDS para definir una región de detección. Ésta es definida como una pequeña franja horizontal localizada por debajo del horizonte. La principal diferencia entre los dos métodos que usan esta simplificación es que en su trabajo más actual en lugar de utilizar la línea del horizonte, esta región es definida 10 píxeles por debajo del horizonte, con el objetivo de poder buscar las regiones en una área con un fondo estructurado simple. Otra forma de reducir el coste computacional del IPM para la detección de obstáculos es mediante la implementación del IPM en un hardware adicional al computador de a bordo como una FPGA [140].

Muchos de los algoritmos que utilizan IPM para la detección de obstáculos calculan el desplazamiento de la plataforma mediante la evaluación de dos imágenes. Por otra parte, otra forma de obtener IPM es mediante un sensor binocular. En el trabajo de Kim et al. [141], se presentan varias aportaciones como una reducción del tiempo computacional que exigen los sensores binoculares para extraer la información, una solución a la variación de iluminación y una obtención de la profundidad más precisa, especialmente para obstáculos móviles. Esto se consigue mediante un método de agregación de costes utilizando información de color y movimiento.

IPM es una de las técnicas que se utilizan en combinación con OF. Por ejemplo, en [142], se utiliza esta combinación para detectar obstáculos en los puntos muertos de los laterales de los vehículos.

Aparte de la detección de obstáculos, IPM también es utilizado en la detección de líneas [143], [136], [144] debido a la capacidad de este método para quitar la perspectiva de la imagen. Esta detección de líneas se puede utilizar como un paso previo a la detección de obstáculos [143]. Otro trabajo que detecta líneas y obstáculos delante de un vehículo es presentado en [145], donde los autores utilizan una combinación de IPM con un sensor binocular.

IV-C. Otros

Como se ha descrito anteriormente, el OF es una técnica muy utilizada en la detección de obstáculos, esta puede utilizarse con el TTC o se puede combinar con otros métodos. En el OF se consideran dos tipos de movimientos, el de rotación y el de traslación. Los movimientos de rotación provocan falsas señales cuando se trata de estimar los vectores de flujo óptico.

En [146] y [147], se presenta un método biológicamente inspirado utilizado en un avión de 30 gramos capaz de detectar obstáculos mediante dos cámaras monoculares. El flujo óptico de traslación se obtiene eliminando los efectos de rotación mediante la información proveniente de un giróscopo. El obstáculo es detectado mediante la comparación del flujo óptico entre la cámara derecha e izquierda. Varios años después, un enfoque similar al propuesto por Zufferey y Floreano [146][147] es presentado en [148], donde los obstáculos son detectados mediante el análisis de los valores del flujo óptico. Los valores que superan un umbral son considerados obstáculos. Este umbral es ajustado según la velocidad y la maniobrabilidad del avión, el ángulo de aproximación, y las características y los ángulos de colocación de la cámara.

Como es bien sabido Horn-Schunck y Lucas-Kanade son unos de los precursores de los algoritmos de flujo óptico. En [149], los autores presentan una comparación de cada uno de estos métodos para la detección de obstáculos.

Por otra parte, existen una gran variedad de algoritmos basados en segmentación. A continuación se detallan un conjunto de métodos capaces de detectar obstáculos o regiones transitables mediante la segmentación del entorno basándose en la información de color.

Existe un grupo de algoritmos que utilizan muy pocos recursos computacionales para segmentar la imagen basándose en el color del terreno. Estos algoritmos simplifican la detección mediante la clasificación de los píxeles de una imagen en obstáculos o regiones transitables. Una de las suposiciones que utilizan estos algoritmos es que los suelos son planos. Esto implica que todos los objetos tienen su base en el suelo y que la distancia del robot al objeto es lineal [150], [151], [152]. En [151], los autores presentan un algoritmo capaz de segmentar el entorno entre zonas transitables y obstáculos utilizando solo una imagen de 22×30 píxeles. Las dimensiones de esta área de evaluación son inferiores a otro algoritmo muy similar [150]. Reduciendo el ciclo computacional se puede utilizar una electrónica más simple para detectar obstáculos. Otro trabajo que utiliza la segmentación basándose en el color del terreno es propuesto en [152]. Estos presentan un método muy similar al de [150]. Las principales diferencias son que en el trabajo de Ulrich y Nourbakhsh [152] no imponen la restricción de que el área de referencia esta siempre libre de obstáculos. Además, éste proporciona imágenes binarias de los obstáculos a alta resolución en tiempo real.

La segmentación basada en color no sólo es utilizada para detectar obstáculos unidos al suelo. En [153], se propone un algoritmo capaz de detectar obstáculos mediante la segmentación del cielo utilizando una sola imagen. Este método segmenta la imagen en regiones de cielo y no cielo mediante la clasificación de sus píxeles utilizando el espacio de color YCrCb. Se calcula la línea que representa el horizonte y que mejor separa la región de cielo y no cielo. A continuación, todos los píxeles que son detectados como regiones de no cielo sobre la línea del horizonte son clasificados como obstáculos.

Otros métodos de detección de obstáculos mediante la

combinación de textura y color son obtenidos mediante una evaluación de las regiones más notorias de la imagen o mediante la identificación de formas. En [154], se utiliza el concepto de que la variación de textura en la imagen es menor cuando el objeto esta más cerca. En [155], los autores describen un algoritmo capaz de detectar obstáculos basado en las regiones notorias de la imagen. Este concepto biológicamente inspirado está ligado a que los humanos somos capaces de detectar obstáculos mediante la identificación de zonas que atraen nuestra atención. Otro método capaz de detectar obstáculos en el entorno es presentado en [156] y se basa en una comparación entre los obstáculos y patrones de los obstáculos a detectar.

V. CONCLUSIONES

En este estado del arte se ha presentado una vista general de los métodos más importantes para la detección de obstáculos y de áreas transitables. Analizando los diferentes métodos citados en este estado del arte, se puede observar que no existe un método universal para todos los entornos, sino que cada uno presenta ventajas e inconvenientes en su utilización en diferentes entornos. Cabe destacar que si se tuviese que elegir un método que se pudiera adaptar a la mayoría de los entornos posibles, éste sería el método de generación de mapas, tanto monocular como binocular, aunque por otra parte, estos necesitan una cierta capacidad computacional para ser ejecutados.

La primera clasificación que se ha hecho en este estado del arte se corresponde a la división entre métodos que calculan la profundidad para la detección de obstáculos y los que no. Esta clasificación es muy importante ya que obteniendo la profundidad se puede conocer la distancia del obstáculo a la plataforma, mientras que si la profundidad no es conocida sólo se puede diferenciar entre regiones ocupadas y libres en la imagen. Otra diferencia entre ambos métodos es que los que no obtienen profundidad generalmente suelen ser menos complejos y necesitan menos recursos computacionales.

Una de las principales restricciones cuando se habla de algoritmos es que se puedan ejecutar en tiempo real. Para conseguir que los algoritmos de generación de mapas se puedan ejecutar en tiempo real, en algunos casos, es necesario el uso de hardware adicional tal como FPGAs, GPUs o procesadores ARM. Dependiendo de la tarea que se va a realizar después de la detección de obstáculos, este conocimiento de la profundidad puede ser un requerimiento esencial o una información no relevante. Dentro de esta primera clasificación se han abordado otros métodos como la generación de mapas por medios tanto monoculares como binoculares, la evaluación del tamaño de los puntos característicos en diferentes instantes de tiempo para detectar su expansión, la utilización de métodos de aprendizaje automático, el uso de la apariencia de la imagen o el mapa de perspectiva inversa.

Dos de los requisitos necesarios a la hora de elegir un método adecuado para la detección de obstáculos son la plataforma a utilizar y el entorno donde se moverá. Existe una gran variedad de entornos, los cuales se pueden clasificar a su vez de muchas formas. En robótica, normalmente se suele clasificar en entornos estructurados, o no

estructurados o entornos de interior y exterior. Además de los entornos citados anteriormente, existen una serie de entornos que es importante diferenciar al respecto de la detección de obstáculos como, por ejemplo, diferenciar entre entornos con o sin suelo. Muchos de los algoritmos utilizados en robótica terrestre necesitan detectar el suelo ya que éste es un paso previo a la detección de obstáculos. En el ámbito de los coches, existen entornos urbanos o autopistas, donde según el entorno será necesario detectar diferentes tipos de obstáculos y a diferentes tasas de video. En robótica aérea, es importante diferenciar entre interior y exterior, ya que existen diferencias entre los obstáculos y el entorno.

Otro factor a la hora de elegir un método de detección de obstáculos es la plataforma que se va a utilizar. Una de los factores a tener en cuenta cuando se analiza una plataforma robótica es su dinámica. En la plataforma con movimientos relativamente lentos la velocidad de procesamiento necesaria es inferior a una que se mueva muy rápido, ya que en esta última el entorno varía rápidamente. Otro factor a tener en cuenta a la hora de elegir un método adecuado son los grados de libertad del vehículo y su dirección habitual de movimiento. Según su movimiento, será necesario obtener información en una sola dirección o en múltiples direcciones. Para detectar obstáculos en una sola dirección será necesario colocar la cámara en una posición específica. Sin embargo, si se quiere detectar muchas direcciones será necesario incorporar una de las siguientes soluciones: utilizar más de una cámara, una cámara omnidireccional o una cámara con un gran ángulo de visión. La resolución de la cámara es otro factor importante: a más resolución, más detalle y mayor capacidad de detectar obstáculos más pequeños, mayor precisión en la detección de objetos más lejanos, pero al mismo tiempo, el requerimiento computacional para procesar toda la imagen es mayor. Otra parte importante de las cámaras son las lentes. Las lentes de ángulo estrecho son utilizadas para la detección de objetos lejanos y poder conseguir un mayor número de píxeles lejanos en la imagen, mientras que para detectar obstáculos a poca distancia es mejor utilizar las lentes de gran angular.

Por otra parte, otros requisitos necesarios para la detección de obstáculos es el conocimiento previo de algunos requisitos como la capacidad computacional disponible y la tarea que se debe realizar posteriormente. Según la tarea a realizar, ésta nos definirá el nivel de detalle y las características necesarias del obstáculo detectado, tal como la profundidad, su localización, si los obstáculos son dinámicos o estáticos, la forma del obstáculo, la identificación de un objeto específico, las regiones libres o con presencia de obstáculos, etc.

Muchos de los métodos presentados en este estudio requieren de odometría o del posicionamiento respecto al entorno para poder almacenar datos a lo largo del tiempo en mapas, o para poder eliminar o compensar movimientos específicos que afectan al procesamiento de la imagen entre instantes de tiempo consecutivos. Otro factor importante tiene que ver con las propiedades dinámicas de los obstáculos. Existen obstáculos estáticos y dinámicos. Generalmente todos los métodos descritos son capaces de

tratar con la detección de obstáculos estáticos, mientras que para la detección de obstáculos dinámicos los métodos disponibles realizan una evaluación en el tiempo de las imágenes. Las técnicas más comunes para realizar esta evaluación son los métodos de seguimiento y de flujo óptico. La mayoría de obstáculos dinámicos que se han visto en este estado del arte son coches y peatones. Aparte de las propiedades dinámicas de los obstáculos, muchos pueden ser identificados mediante su forma y color. Por eso muchos métodos utilizan la apariencia del obstáculo a detectar o para crear un modelo.

AGRADECIMIENTOS

Este trabajo está parcialmente financiado por la beca BES-2015-071804 (proyecto DPI2014-57746-C3-2-R) y por el proyecto EU-H2020 ROBINS (GA779776).

REFERENCIAS

- [1] F. Bonin-Font, A. Ortiz, and G. Oliver, "Visual navigation for mobile robots: A survey," *Journal of Intelligent and Robotic Systems*, vol. 53, no. 3, pp. 263–296, 2008.
- [2] I. Kostavelis and A. Gasteratos, "Semantic mapping for mobile robotics tasks: A survey," *Robotics and Autonomous Systems*, vol. 66, pp. 86 – 103, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0921889014003030>
- [3] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *Acm computing surveys*, vol. 38, no. 4, p. 13, 2006.
- [4] N. Bernini, M. Bertozzi, L. Castangia, M. Patander, and M. Sabbatelli, "Real-time obstacle detection using stereo vision for autonomous ground vehicles: A survey," in *IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2014, pp. 873–878.
- [5] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.
- [6] M. Enzweiler and D. M. Gavrilu, "Monocular pedestrian detection: Survey and experiments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2179–2195, 2009.
- [7] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1239–1258, 2010.
- [8] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1773–1795, 2013.
- [9] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 5, pp. 694–711, 2006.
- [10] T. Gandhi, S. Devadiga, R. Kasturi, and O. Camps, "Detection of obstacles on runways using ego-motion compensation and tracking of significant features," *Image and Vision Computing*, vol. 18, no. 10, pp. 805–815, 2000.
- [11] S. Garrido-Jurado, R. M. noz Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280 – 2292, 2014. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0031320314000235>
- [12] M. Fiala, "Artag, a fiducial marker system using digital techniques," in *International Conference on Computer Vision and Pattern Recognition*, ser. CVPR '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 590–596. [Online]. Available: <http://dx.doi.org/10.1109/CVPR.2005.74>

- [13] J. Pestana, J. L. Sanchez-Lopez, P. de la Puente, A. Carrio, and P. Campoy, "A vision-based quadrotor swarm for the participation in the 2013 international micro air vehicle competition," in *International Conference on Unmanned Aircraft Systems*, May 2014, pp. 617–622.
- [14] L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, "Pixhawk: A system for autonomous flight using onboard computer vision," in *IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 2992–2997.
- [15] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," 1981, pp. 674–679.
- [16] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, no. 1, pp. 185 – 203, 1981. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0004370281900242>
- [17] R. Nelson and J. Aloimonos, "Obstacle avoidance using flow field divergence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 10, pp. 1102–1106, 1989.
- [18] S. S. Beauchemin and J. L. Barron, "The computation of optical flow," *ACM Comput. Surv.*, vol. 27, no. 3, pp. 433–466, Sep. 1995. [Online]. Available: <http://doi.acm.org/10.1145/212094.212141>
- [19] H. Chao, Y. Gu, and M. Napolitano, "A survey of optical flow techniques for robotics navigation applications," *Journal of Intelligent and Robotic Systems*, vol. 73, no. 1, pp. 361–372, 2013. [Online]. Available: <http://dx.doi.org/10.1007/s10846-013-9923-6>
- [20] A. Mendes, L. C. Bento, and U. Nunes, "Multi-target detection and tracking with a laser scanner," in *IEEE Intelligent Vehicles Symposium*, June 2004, pp. 796–801.
- [21] M. Bertozzi, A. Broggi, C. Caraffi, M. Del Rose, M. Felisa, and G. Vezzoni, "Pedestrian detection by means of far-infrared stereo vision," *Computer vision and image understanding*, vol. 106, no. 2, pp. 194–204, 2007.
- [22] F. Suard, A. Rakotomamonjy, A. Bensrhair, and A. Broggi, "Pedestrian detection using infrared images and histograms of oriented gradients," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2006, pp. 206–212.
- [23] S. J. Krotosky and M. M. Trivedi, "On color-, infrared-, and multimodal-stereo approaches to pedestrian detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 4, pp. 619–629, 2007.
- [24] F. Bonnini-Pascual, A. Ortiz, E. Garcia-Fidalgo, and J. P. Company, "A micro-aerial platform for vessel visual inspection based on supervised autonomy," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2015, pp. 46–52.
- [25] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. R. Bradski, "Self-supervised monocular road detection in desert terrain," in *Robotics: science and systems*, vol. 38. Philadelphia, 2006.
- [26] R. Labayrade, C. Royere, D. Gruyer, and D. Aubert, "Cooperative fusion for multi-obstacles detection with use of stereovision and laser scanner," *Autonomous Robots*, vol. 19, no. 2, pp. 117–140, 2005. [Online]. Available: <http://dx.doi.org/10.1007/s10514-005-0611-7>
- [27] A. Discant, A. Rogozan, C. Rusu, and A. Bensrhair, "Sensors for obstacle detection-a survey," in *International Spring Seminar on Electronics Technology*. IEEE, 2007, pp. 100–105.
- [28] S. Thrun *et al.*, "Robotic mapping: A survey," *Exploring artificial intelligence in the new millennium*, vol. 1, pp. 1–35, 2002.
- [29] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: An efficient probabilistic 3d mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, 2013.
- [30] D. Liyanage and M. Perera, "Optical flow based obstacle avoidance for the visually impaired," in *IEEE Business Engineering and Industrial Applications Colloquium*. IEEE, 2012, pp. 284–289.
- [31] E. M. d. O. Caldeira, H. J. A. Schneebeli, and M. Sarcinelli-Filho, "An optical flow-based sensing system for reactive mobile robot navigation," *Sba: Controle & Automação Sociedade Brasileira de Automatica*, vol. 18, no. 3, pp. 265–277, 2007.
- [32] K. Souhila and A. Karim, "Optical flow based robot obstacle avoidance," *International Journal of Advanced Robotic Systems*, vol. 4, no. 1, pp. 13–16, 2007.
- [33] M. S. Guzel and R. Bicker, "Optical flow based system design for mobile robots," in *IEEE Conference on Robotics, Automation and Mechatronics*, June 2010, pp. 545–550.
- [34] Y. Shen, X. Du, and J. Liu, "Monocular vision based obstacle detection for robot navigation in unstructured environment," in *International Symposium on Neural Networks*. Springer, 2007, pp. 714–722.
- [35] C. Wang, W. Liu, and M. Q.-H. Meng, "Obstacle avoidance for quadrotor using improved method based on optical flow," in *IEEE International Conference on Information and Automation*. IEEE, 2015, pp. 1674–1679.
- [36] A. Chavez and D. Gustafson, "Vision-based obstacle avoidance using sift features," in *International Symposium on Visual Computing*. Springer, 2009, pp. 550–557.
- [37] T. Mori and S. Scherer, "First results in detecting and avoiding frontal obstacles from a monocular camera for micro unmanned aerial vehicles," in *IEEE International Conference on Robotics and Automation*. IEEE, 2013.
- [38] G. Alenyà, A. Nègre, and J. L. Crowley, "Time To Contact for Obstacle Avoidance," in *European Conference on Mobile Robotics*, Dubrovnik, Croatia, Sep. 2009. [Online]. Available: <https://hal.inria.fr/inria-00426735>
- [39] D. Muller, J. Pauli, C. Nunn, S. Gormer, and S. Muller-Schneiders, "Time to contact estimation using interest points," in *IEEE International Conference on Intelligent Transportation Systems*, Oct 2009, pp. 1–6.
- [40] G. Alenyà, A. Nègre, and J. L. Crowley, "A comparison of three methods for measure of time to contact," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2009, pp. 4565–4570.
- [41] S. Pundlik, E. Peli, and G. Luo, "Time to collision and collision risk estimation from local scale and motion," in *Advances in Visual Computing*. Springer, 2011, pp. 728–737.
- [42] C. Bills, J. Chen, and A. Saxena, "Autonomous mav flight in indoor environments using single image perspective cues," in *IEEE International Conference on Robotics and Automation*, 2011. [Online]. Available: <http://mav.cs.cornell.edu/>
- [43] Y. S. Liao, Q. Zhang, Y. Li, and S. S. Ge, "Non-metric navigation for mobile robot using optical flow," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 4953–4958.
- [44] E. Martinez, M. Diaz, J. Melenchon, J. A. Montero, I. Iriondo, and J. C. Socoro, "Driving assistance system based on the detection of head-on collisions," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2008, pp. 913–918.
- [45] B. K. P. Horn, Y. Fang, and I. Masaki, "Time to contact relative to a planar surface," in *IEEE Intelligent Vehicles Symposium*, June 2007, pp. 68–74.
- [46] A. Nolan, D. Serrano, A. H. Sabate, D. Ponsa-Mussarra, and A. Lopez-Pena, "Obstacle mapping module for quadrotors on outdoor search and rescue operations," in *International Micro Air Vehicle Conference and Flight Competition*, 2013.
- [47] C. Hane, T. Sattler, and M. Pollefeys, "Obstacle detection for self-driving cars using only monocular cameras and wheel odometry," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2015, pp. 5101–5108.
- [48] B. Call, R. Beard, C. Taylor, and B. Barber, "Obstacle avoidance for unmanned air vehicles using image feature tracking," in *AIAA Guidance, Navigation, and Control Conference*, 2006, pp. 3406–3414.

- [49] S. Weiss, M. Achtelik, L. Kneip, D. Scaramuzza, and R. Siegwart, "Intuitive 3d maps for mav terrain exploration and obstacle avoidance," *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1-4, pp. 473-493, 2011.
- [50] R. Roberts, D.-N. Ta, J. Straub, K. Ok, and F. Dellaert, "Saliency detection and model-based tracking: a two part vision system for small robot navigation in forested environments," in *SPIE Defense, Security, and Sensing*, vol. 8387, 2012, pp. 83 870S-1.
- [51] A. Wedel, U. Franke, J. Klappstein, T. Brox, and D. Cremers, "Realtime depth estimation and obstacle detection from monocular video," in *Joint Pattern Recognition Symposium*. Springer, 2006, pp. 475-484.
- [52] S. Zingg, D. Scaramuzza, S. Weiss, and R. Siegwart, "Mav navigation through indoor corridors using optical flow," in *IEEE International Conference on Robotics and Automation*. IEEE, 2010, pp. 3361-3368.
- [53] V. Lippiello, G. Loianno, and B. Siciliano, "Mav indoor navigation based on a closed-form solution for absolute scale velocity estimation using optical flow and inertial data," in *IEEE Conference on Decision and Control and European Control*. IEEE, 2011, pp. 3566-3571.
- [54] S. Hrabar and G. S. Sukhatme, "A comparison of two camera configurations for optic-flow based navigation of a uav through urban canyons," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, Sept 2004, pp. 2673-2680 vol.3.
- [55] J. Serres, D. Dray, F. Ruffier, and N. Franceschini, "A vision-based autopilot for a miniature air vehicle: joint speed control and lateral obstacle avoidance," *Autonomous Robots*, vol. 25, no. 1, pp. 103-122, 2007. [Online]. Available: <http://dx.doi.org/10.1007/s10514-007-9069-0>
- [56] J. Stowers, M. Hayes, and A. Bainbridge-Smith, "Biologically inspired uav obstacle avoidance and control using monocular optical flow & divergence templates," in *International Conference on Automation, Robotics and Applications*. IEEE, 2011, pp. 378-383.
- [57] J. Michels, A. Saxena, and A. Y. Ng, "High speed obstacle avoidance using monocular vision and reinforcement learning," in *International Conference on Machine Learning*, 2005. [Online]. Available: <http://www.cs.cornell.edu/~asaxena/rccar/>
- [58] A. Saxena, S. H. Chung, and A. Y. Ng, "Learning depth from single monocular images," in *Advances in Neural Information Processing Systems*, 2005, pp. 1161-1168.
- [59] R. Kalarot and J. Morris, "Comparison of fpga and gpu implementations of real-time stereo vision," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2010, pp. 9-15.
- [60] A. J. Barry, H. Oleynikova, D. Honegger, M. Pollefeys, and R. Tedrake, "Fast onboard stereo vision for uavs," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015.
- [61] K. Schmid, T. Tomic, F. Ruess, H. Hirschmuller, and M. Suppa, "Stereo vision based indoor/outdoor navigation for flying robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 3955-3962.
- [62] S. Hrabar, "3d path planning and stereo-based obstacle avoidance for rotorcraft uavs," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 807-814.
- [63] L. Heng, L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, "Autonomous obstacle avoidance and maneuvering on a vision-guided mav using on-board processing," in *IEEE International Conference on Robotics and Automation*, 2011.
- [64] F. Andert, F. Adolf, L. Goormann, and J. Dittrich, "Mapping and path planning in complex environments: An obstacle avoidance approach for an unmanned helicopter," in *IEEE International Conference on Robotics and Automation*, May 2011, pp. 745-750.
- [65] P. Gohl, D. Honegger, S. Omari, M. Achtelik, M. Pollefeys, and R. Siegwart, "Omnidirectional visual obstacle detection using embedded fpga," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2015. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7353931
- [66] H. Lategahn, W. Derendarz, T. Graf, B. Kitt, and J. Effertz, "Occupancy grid computation from dense stereo and sparse structure and motion points for automotive applications," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2010, pp. 819-824.
- [67] H. Wang and Z. Wei, "Stereovision based obstacle detection system for unmanned surface vehicle," in *IEEE International Conference on Robotics and Biomimetics*. IEEE, 2013, pp. 917-921.
- [68] H. Wang, Z. Wei, S. Wang, C. S. Ow, K. T. Ho, and B. Feng, "A vision-based obstacle detection system for unmanned surface vehicle," in *IEEE Robotics and Automation Magazine*. IEEE, 2011, pp. 364-369.
- [69] A. Broggi, S. Cattani, M. Patander, M. Sabbatelli, and P. Zani, "A full-3d voxel-based dynamic obstacle detection for urban scenario using stereo vision," in *IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2013, pp. 71-76.
- [70] C. Brailion, C. Pradalier, K. Usher, J. L. Crowley, and C. Laugier, "Occupancy grids from stereo and optical flow data," in *Experimental Robotics*. Springer, 2008, pp. 367-376.
- [71] C. D. Pantilie and S. Nedevschi, "Real-time obstacle detection in complex scenarios using dense stereo vision and optical flow," in *IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2010, pp. 439-444.
- [72] F. Oniga and S. Nedevschi, "Processing dense stereo data using elevation maps: Road surface, traffic isle, and obstacle detection," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 3, pp. 1172-1182, 2010.
- [73] R. Danescu and S. Nedevschi, "A particle-based solution for modeling and tracking dynamic digital elevation maps," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 3, pp. 1002-1015, 2014.
- [74] R. Danescu, F. Oniga, and S. Nedevschi, "Modeling and tracking the driving environment with a particle-based occupancy grid," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1331-1342, 2011.
- [75] R. Bichsel and P. V. K. Borges, "Discrete-continuous clustering for obstacle detection using stereo vision," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2015, pp. 538-545.
- [76] H. Badino, U. Franke, and D. Pfeiffer, *The Stixel World - A Compact Medium Level Representation of the 3D-World*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 51-60. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-03798-6_6
- [77] R. Benenson, R. Timofte, and L. Van Gool, "Stixels estimation without depth map computation," in *International Conference on Computer Vision*. IEEE, 2011, pp. 2010-2017.
- [78] A. Wedel, U. Franke, H. Badino, and D. Cremers, "B-spline modeling of road surfaces for freespace estimation," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2008, pp. 828-833.
- [79] M. Muffert, N. Schneider, and U. Franke, "Stix-fusion: a probabilistic stixel integration technique," in *Canadian Conference on Computer and Robot Vision*. IEEE, 2014, pp. 16-23.
- [80] M. Muffert, S. Anzt, and U. Franke, "An incremental map building approach via static stixel integration," *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, 2013.
- [81] M. Cordts, L. Schneider, M. Enzweiler, U. Franke, and S. Roth, "Object-level priors for stixel generation," in *German Conference on Pattern Recognition*. Springer, 2014, pp. 172-183.
- [82] D. Pfeiffer and U. Franke, "Efficient representation of traffic scenes by means of dynamic stixels," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2010, pp. 217-224.

- [83] F. Erbs, B. Schwarz, and U. Franke, "Stixmentation-probabilistic stixel based traffic scene labeling," in *British Machine Vision Conference*, 2012, pp. 1–12.
- [84] H. Oleynikova, D. Honegger, and M. Pollefeys, "Reactive avoidance using embedded stereo vision for mav flight," in *IEEE International Conference on Robotics and Automation*. IEEE, 2015. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7138979
- [85] R. Labayrade, D. Aubert, and J. P. Tarel, "Real time obstacle detection in stereovision on non flat road geometry through v-disparity representation," in *IEEE Intelligent Vehicles Symposium*, 2002.
- [86] Z. Hu and K. Uchimura, "Uv-disparity: an efficient algorithm for stereovision based scene analysis," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2005, pp. 48–54.
- [87] N. Soquet, D. Aubert, and N. Hautiere, "Road segmentation supervised by an extended v-disparity algorithm for autonomous navigation," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2007, pp. 160–165.
- [88] J. Zhao, M. Whitty, and J. Katupitiya, "Detection of non-flat ground surfaces using v-disparity images," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2009, pp. 4584–4589.
- [89] G. De Cubber, D. Doroftei, L. Nalpantidis, G. C. Sirakoulis, and A. Gasteratos, "Stereo-based terrain traversability analysis for robot navigation," in *IARP/EURON Workshop on Robotics for Risky Interventions and Environmental Surveillance*, Brussels, Belgium, 2009.
- [90] X. Zhu, H. Lu, X. Yang, Y. Li, and H. Zhang, "Stereo vision based traversable region detection for mobile robots using uv-disparity," in *Chinese Control Conference*. IEEE, 2013, pp. 5785–5790.
- [91] M. Bai, Y. Zhuang, and W. Wang, "Stereovision based obstacle detection approach for mobile robot navigation," in *International Conference on Intelligent Control and Information Processing*, Aug 2010, pp. 328–333.
- [92] Z. Hu, F. Lamosa, and K. Uchimura, "A complete uv-disparity study for stereovision based 3d driving environment analysis," in *International Conference on 3-D Digital Imaging and Modeling*. IEEE, 2005, pp. 204–211.
- [93] M. Gouiffes, A. Patri, and M. Vasiliu, "Robust obstacles detection and tracking using disparity for car driving assistance," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2010, pp. 75 390H–75 390H.
- [94] C. Caraffi, S. Cattani, and P. Grisleri, "Off-road path and obstacle detection using decision networks and stereo vision," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 4, pp. 607–618, 2007.
- [95] C. Fu, A. Carrio, and P. Campoy, "Efficient visual odometry and mapping for unmanned aerial vehicle using arm-based stereo vision pre-processing system," in *International Conference on Unmanned Aircraft Systems*, 2015, pp. 957–962.
- [96] A. Broggi, M. Buzzoni, M. Felisa, and P. Zani, "Stereo obstacle detection in challenging environments: the viac experience," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 1599–1604.
- [97] C. Brand, M. J. Schuster, H. Hirschmuller, and M. Suppa, "Stereo-vision based obstacle mapping for indoor/outdoor SLAM," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 1846–1853.
- [98] Z. Xiaozhou, L. Huimin, Y. Xingrui, L. Yubo, and Z. Hui, "Stereo vision based traversable region detection for mobile robots using uv-disparity," in *Chinese Control Conference*. IEEE, 2013, pp. 5785–5790.
- [99] R. Moore, S. Thurrowgood, D. Bland, D. Soccol, and M. Srinivasan, "A stereo vision system for uav guidance," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 3386–3391.
- [100] P. Fleischmann and K. Berns, "A stereo vision based obstacle detection system for agricultural applications," in *International Conference on Field and Service Robots*. Springer, 2016, pp. 217–231.
- [101] S. Hrabar, G. Sukhatme, P. Corke, K. Usher, and J. Roberts, "Combined optic-flow and stereo-based navigation of urban canyons for a uav," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 3309–3316.
- [102] A. Wedel, A. Meißner, C. Rabe, U. Franke, and D. Cremers, "Detection and segmentation of independently moving objects from dense scene flow," in *Energy minimization methods in computer vision and pattern recognition*. Springer, 2009, pp. 14–27.
- [103] P. Lenz, J. Ziegler, A. Geiger, and M. Roser, "Sparse scene flow segmentation for moving object detection in urban environments," in *IEEE Intelligent Vehicles Symposium*, 2011. [Online]. Available: <http://rainsoft.de/publications.html>
- [104] S. Tijmons, G. C. de Croon, B. D. Remes, C. De Wagter, and M. Mulder, "Obstacle avoidance strategy using onboard stereo vision on a flapping wing mav," *IEEE Transactions on Robotics*, 2017.
- [105] P. Pinggera, U. Franke, and R. Mester, "High-performance long range obstacle detection using stereo vision," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2015, pp. 1308–1313.
- [106] S. Heinrich, "Real time fusion of motion and stereo using flow/depth constraint for fast obstacle detection," in *Joint Pattern Recognition Symposium*. Springer, 2002, pp. 75–82.
- [107] T. Vaudrey, A. Wedel, C. Rabe, J. Klappstein, and R. Klette, "Evaluation of moving object segmentation comparing 6d-vision and monocular motion constraints," in *International Conference Image and Vision Computing*. IEEE, 2008, pp. 1–6.
- [108] P. Pinggera, U. Franke, and R. Mester, "Highly accurate depth estimation for objects at large distances," in *German Conference on Pattern Recognition*. Springer, 2013, pp. 21–30.
- [109] C. Rabe, U. Franke, and S. Gehrig, "Fast detection of moving objects in complex scenarios," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2007, pp. 398–403.
- [110] U. Franke, C. Rabe, H. Badino, and S. Gehrig, "6d-vision: Fusion of stereo and motion for robust environment perception," in *Pattern Recognition*. Springer, 2005, pp. 216–223.
- [111] A. Talukder and L. Matthies, "Real-time detection of moving objects from moving vehicles using dense stereo and optical flow," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 4. IEEE, 2004, pp. 3718–3725.
- [112] A. Ess, K. Schindler, B. Leibe, and L. Van Gool, "Object detection and tracking for autonomous navigation in dynamic environments," *International Journal of Robotics Research*, vol. 29, no. 14, pp. 1707–1725, 2010.
- [113] H. Badino, U. Franke, and R. Mester, "Free space computation using stochastic occupancy grids and dynamic programming," in *International Conference on Computer Vision*, vol. 20, 2007.
- [114] G. Toulminet, M. Bertozzi, S. Mousset, A. Bensrhair, and A. Broggi, "Vehicle detection by means of stereo vision-based obstacles features extraction and monocular pattern analysis," *IEEE Transactions on Image Processing*, vol. 15, no. 8, pp. 2364–2375, 2006.
- [115] R. Hadsell, P. Sermanet, J. Ben, A. Erkan, M. Scoffier, K. Kavukcuoglu, U. Muller, and Y. LeCun, "Learning long-range vision for autonomous off-road driving," *Journal of Field Robotics*, vol. 26, no. 2, pp. 120–144, 2009.
- [116] I. Lenz, M. Gemici, and A. Saxena, "Low-power parallel algorithms for single image based obstacle avoidance in aerial robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 772–779.
- [117] M. Kristan, V. S. Kenk, S. Kovačič, and J. Perš, "Fast image-based obstacle detection from unmanned surface vehicles," *IEEE transactions on cybernetics*, vol. 46, no. 3, pp. 641–654, 2016.

- [118] G. De Croon, C. De Wagter, B. Remes, and R. Ruijsink, "Sky segmentation approach to obstacle avoidance," in *IEEE Aerospace Conference*. IEEE, 2011, pp. 1–16.
- [119] P. Alcantarilla, L. Bergasa, P. Jimenez, M. Sotelo, I. Parra, D. Fernandez, and S. Mayoral, "Night time vehicle detection for driving assistance lightbeam controller," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2008, pp. 291–296.
- [120] A. Maldonado-Ramirez, L. A. Torres-Mendez, and F. Rodriguez-Telles, "Ethologically inspired reactive exploration of coral reefs with collision avoidance: Bridging the gap between human and robot spatial understanding of unstructured environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2015, pp. 4872–4879.
- [121] D. Kim, J. Sun, S. M. Oh, J. M. Rehg, and A. F. Bobick, "Traversability classification using unsupervised on-line visual learning for outdoor robot navigation," in *IEEE International Conference on Robotics and Automation*. IEEE, 2006, pp. 518–525.
- [122] D. Lieb, A. Lookingbill, and S. Thrun, "Adaptive road following using self-supervised learning and reverse optical flow," in *Robotics: Science and Systems*, 2005, pp. 273–280.
- [123] M. Blanchard, F. C. Rind, and P. F. Verschure, "Collision avoidance using a model of the locust lgmd neuron," *Robotics and Autonomous Systems*, vol. 30, no. 1, pp. 17–38, 2000.
- [124] S. Yue, F. C. Rind, M. S. Keil, J. Cuadri, and R. Stafford, "A bio-inspired visual collision detection mechanism for cars: Optimisation of a model of a locust neuron to a novel environment," *Neurocomputing*, vol. 69, no. 13, pp. 1591–1598, 2006.
- [125] J. Cuadri, G. Linan, R. Stafford, M. Keil, and E. Roca, "A bio-inspired collision detection algorithm for vlsi implementation," in *Microtechnologies for the New Millennium*. International Society for Optics and Photonics, 2005, pp. 238–248.
- [126] R. Stafford, R. D. Santer, and F. C. Rind, "A bio-inspired visual collision detection mechanism for cars: combining insect inspired neurons to create a robust system," *BioSystems*, vol. 87, no. 2, pp. 164–171, 2007.
- [127] H. Okuno and T. Yagi, "Real-time robot vision for collision avoidance inspired by neuronal circuits of insects," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2007, pp. 1302–1307.
- [128] S. B. i Badia and P. F. Verschure, "A collision avoidance model based on the lobula giant movement detector (lgmd) neuron of the locust," in *International Joint Conference on Neural Networks*, vol. 3. IEEE, 2004, pp. 1757–1761.
- [129] S. B. i Badia, P. Pyk, and P. F. Verschure, "A fly-locust based neuronal control system applied to an unmanned aerial vehicle: the invertebrate neuronal principles for course stabilization, altitude control and collision avoidance," *International Journal of Robotics Research*, vol. 26, no. 7, pp. 759–772, 2007.
- [130] H. A. Mallot, H. H. Bülthoff, J. Little, and S. Bohrer, "Inverse perspective mapping simplifies optical flow computation and obstacle detection," *Biological cybernetics*, vol. 64, no. 3, pp. 177–185, 1991.
- [131] M. Bertozzi, A. Broggi, and A. Fascioli, "Stereo inverse perspective mapping: theory and applications," *Image and Vision Computing*, vol. 16, no. 8, pp. 585–590, 1998.
- [132] F. Bonin-Font, A. Burguera, A. Ortiz, and G. Oliver, "Concurrent visual navigation and localisation using inverse perspective transformation," *Electronics letters*, vol. 48, no. 5, pp. 264–266, 2012.
- [133] —, "A monocular mobile robot reactive navigation approach based on the inverse perspective transformation," *Robotica*, vol. 31, no. 02, pp. 225–249, 2013.
- [134] F. Bonin-Font, J. A. Tobaruela, A. O. Rodriguez, and G. Oliver, "Vision-based mobile robot motion control combining t 2 and nd approaches," *Robotica*, vol. 32, no. 04, pp. 591–609, 2014.
- [135] M. Nieto, L. Salgado, F. Jaureguizar, and J. Cabrera, "Stabilization of inverse perspective mapping images based on robust vanishing point estimation," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2007, pp. 315–320.
- [136] P. Cerri and P. Grisleri, "Free space detection on highways using time correlation between stabilized sub-pixel precision ipm images," in *IEEE International Conference on Robotics and Automation*. IEEE, 2005, pp. 2223–2228.
- [137] Z. Yankun, C. Hong, and N. Weyrich, "A single camera based rear obstacle detection system," in *IEEE Intelligent Vehicles Symposium*. IEEE, 2011, pp. 485–490.
- [138] G. Ma, S.-B. Park, S. Muller-Schneiders, A. Ioffe, and A. Kummert, "Vision-based pedestrian detection-reliable pedestrian candidate detection by combining ipm and a 1d profile," in *IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2007, pp. 137–142.
- [139] G. Ma, D. Muller, S.-B. Park, S. Muller-Schneiders, and A. Kummert, "Pedestrian detection using a single monochrome camera," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, no. 1, pp. 42–56, 2009.
- [140] A. Alhamwi, B. Vandeportale, and J. Piat, "Real time vision system for obstacle detection and localization on fpga," in *European Conference on Computer Vision*. Springer, 2015, pp. 80–90.
- [141] D. Kim, J. Choi, H. Yoo, U. Yang, and K. Sohn, "Rear obstacle detection system with fisheye stereo camera using hct," *Expert Systems with Applications*, vol. 42, no. 17, pp. 6295–6305, 2015.
- [142] S. Tan, J. Dale, A. Anderson, and A. Johnston, "Inverse perspective mapping and optic flow: A calibration method and a quantitative analysis," *Image and Vision Computing*, vol. 24, no. 2, pp. 153–165, 2006.
- [143] G. Y. Jiang, T. Y. Choi, S. K. Hong, J. W. Bae, and B. S. Song, "Lane and obstacle detection based on fast inverse perspective mapping algorithm," in *International Conference on Systems, Man, and Cybernetics*, vol. 4. IEEE, 2000, pp. 2969–2974.
- [144] A. M. Muad, A. Hussain, S. A. Samad, M. M. Mustafa, and B. Y. Majlis, "Implementation of inverse perspective mapping algorithm for the development of an automatic lane tracking system," in *TENCON IEEE Region Conference*. IEEE, 2004, pp. 207–210.
- [145] M. Bertozzi and A. Broggi, "Gold: A parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE transactions on image processing*, vol. 7, no. 1, pp. 62–81, 1998.
- [146] J. C. Zufferey and D. Floreano, "Toward 30-gram autonomous indoor aircraft: Vision-based obstacle avoidance and altitude control," in *IEEE International Conference on Robotics and Automation*, April 2005, pp. 2594–2599.
- [147] J.-C. Zufferey and D. Floreano, "Fly-inspired visual steering of an ultralight indoor aircraft," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 137–146, 2006.
- [148] M. Rezaei and F. Saghaei, "Optical flow-based obstacle avoidance of a fixed-wing mav," *Aircraft Engineering and Aerospace Technology*, vol. 83, no. 2, pp. 85–93, 2011. [Online]. Available: <http://dx.doi.org/10.1108/00022661111120971>
- [149] G. Recchia, G. Fasano, D. Accardo, A. Moccia, and L. Paparone, "An optical flow based electro-optical see-and-avoid system for uavs," in *IEEE Aerospace Conference*. IEEE, 2007, pp. 1–9.
- [150] L. M. Lorigo, R. A. Brooks, and W. Grimsou, "Visually-guided obstacle avoidance in unstructured environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1. IEEE, 1997, pp. 373–379.
- [151] C. N. Viet and I. Marshall, "Vision-based obstacle avoidance for a small, low-cost robot," in *International Conference on Advanced Robotics*, 2007.
- [152] I. Ulrich and I. Nourbakhsh, "Appearance-based obstacle detection with monocular color vision," in *AAAI/IAAI*, 2000, pp. 866–871.
- [153] T. G. McGee, R. Sengupta, and K. Hedrick, "Obstacle detection for small autonomous aircraft using sky segmentation," in

IEEE International Conference on Robotics and Automation.
IEEE, 2005, pp. 4679–4684.

- [154] G. De Croon, E. De Weerdt, C. De Wagter, B. Remes, and R. Ruijsink, “The appearance variation cue for obstacle avoidance,” *IEEE Transactions on Robotics and Automation*, vol. 28, no. 2, pp. 529–534, 2012.
- [155] Z. Ma, T. Hu, L. Shen, W. Kong, and B. Zhao, “A detection and relative direction estimation method for uav in sense-and-avoid,” in *IEEE International Conference on Information and Automation*, 2015, pp. 2677–2682.
- [156] F. Wen, Z. Qu, C. Wang, and B. Hu, “Study on real-time obstacle avoidance of mobile robot based on vision sensor,” in *IEEE International Conference on Automation and Logistics*. IEEE, 2008, pp. 2438–2442.