Memorias del XVI Congreso Latinoamericano
de Control Automático, CLCA 2014
Octubre 14-17, 2014. Cancún, Quintana Roo, México

Identificación de objetos mediante un Quadrotor equipado con una cámara y redes neuronales

Gerardo Hernández-H, Humberto Sossa, Elsa Rubio, Juan A. Escareño*

Instituto Politécnico Nacional, IPN, Centro de Investigación en Computación, CIC
Av. Juan de Dios Bátiz s/n, Col. Nueva Industrial Vallejo, Del. G.A.M. C.P. 07738, México D.F.
*Institut Polytechnique des Sciences Avancées; 7-9 rue M. Grandcoing, 94200 Ivry-sur-Seine, France.
(e-mail: ghernandez_a13@sagitario.cic.ipn.mx, {hsossa, erubio}@cic.ipn.mx, jantonio.escareno@gmail.com).

Resumen: En el presente trabajo se realiza la clasificación e identificación de varios objetos en imágenes tomadas por un Quadrotor en línea, tomando como base un estudio comparativo de clasificación entre diferentes redes neuronales. Se compararán las Redes Neuronales de Base Radial (RBNN), las Redes Neuronales de Perceptrones Multicapa (MLP), las Máquinas de Vector Soporte (SVM) con núcleo lineal, polinomial y de base radial, así como algunos clasificadores de tipo estadístico (Red Bayesiana, KNN). Se presenta también la aplicación de éstas metodologías a la clasificación de patrones y al reconocimiento de objetos mediante la teoría de puntos de interés.

Palabras Clave: Identificación de objetos, Rasgos descriptores, Redes neuronales, Quadrotor.

1. INTRODUCCIÓN

En general, podemos decir que una imagen es un conjunto de pixeles y que de forma abstracta representa objetos del mundo real. Esta imagen a su vez, puede representar personas, animales, paisajes, etc. Además, es bien sabido que el procesamiento digital de imágenes involucra el uso de la computadora para cambiar y entender la naturaleza de una imagen digital; lo anterior se puede ejemplificar con la detección de los puntos de interés (Rafael C. González, Richard E. Woods, 2002). Los puntos de interés se definen como las partes "interesantes" en una imagen y sus características son utilizadas como punto de partida para varios algoritmos en el área de visión por computadora. Debido a que estas características son utilizadas como las primitivas principales para varios algoritmos, estos últimos serán eficientes, tanto como lo sea el detector/extractor de características. Es así que, una propiedad indispensable para que un detector de puntos de interés aporte buenos resultados, es que exista repetitividad de esos puntos en dos o más imágenes diferentes de la misma escena (Rafael C. González, Richard E. Woods, 2002). La detección de puntos de interés es una operación a bajo nivel en el procesamiento de imágenes, ésta es usualmente la primera operación que se realiza sobre la imagen y examina cada píxel contenido en la imagen, para verificar si existe alguna característica en particular. Si la detección de puntos de interés es parte de otro algoritmo, entonces, este último, típicamente examinará la imagen en la región donde se detectó el punto de interés.

Como prerrequisito para la detección de puntos de interés, la imagen es usualmente suavizada por un núcleo Gaussiano en una representación de espacio a escala (Donghoon et al, 2006). Los puntos de interés proveen de una descripción complementaria de la estructura de la imagen en términos de regiones; ésta descripción complementaria es opuesta al detector de esquinas que es más enfocado a puntos locales (H. Bay 2006). Los descriptores de los puntos de interés contienen frecuentemente un punto de preferencia (un máximo local o centro de gravedad). Los detectores de puntos de interés pueden detectar áreas suavizadas en una imagen, en la cual los detectores de bordes no pueden.

Existen varios algoritmos para la extracción de descriptores/características basados en la detección de puntos de interés; ejemplo de ellos son Scale Invariant Feature Transform (SIFT) y Speed Up Robust Features (SURF), (H. Bay 2006, D. Lowee 1999). En nuestro estudio utilizamos SURF como extractor y detector de puntos de interés, debido a que este posee una dimensionalidad de 64 descriptores por punto de interés detectado, lo cual es menor a los 128 descriptores por punto de interés detectado que nos proporciona el algoritmo SIFT. Esto conlleva una mejora en la velocidad de procesamiento y de una mayor repetitividad de descriptores en imágenes sucesivas de la misma escena.

2. ANTECEDENTES

A continuación se describen las técnicas y algoritmos fundamentales en las que se basa el presente trabajo.

2.1 Imágenes Integrales

Este tipo de cálculo es utilizado para un cómputo rápido de filtros de convolución de tipo caja (H. Bay, 2006). De tal forma

que la entrada de una imagen integral $I_{\sum}(\mathbf{x})$ en el punto $\mathbf{x}=(x,y)$, representa la suma de todos los pixeles de la imagen de entrada I en una región rectangular formada por el punto de origen y el punto $\mathbf{x}$. Su cálculo está dado por la siguiente ecuación:

$$I_{\sum}(\mathbf{x})=\sum_{i=0}^{i\le x}\sum_{j=0}^{j\le y} I(i,j) \qquad (1)$$

Dónde $i,j\in\mathbf{Z}$.

2.2 Scale-Invariant Feature Transform (SIFT)

En 2004, D. Lowe propuso este nuevo algoritmo (D. Lowe 2004), en el cual se extraen puntos característicos de una imagen y se calculan sus descriptores. En el proceso de filtrado en el espacio de escala, Lowe utiliza una Diferencia de Gaussianas (DoG) para aproximar al Laplaciano del Gaussiano (LoG), que es, para diferentes valores de escalas σ, más costoso computacionalmente hablando. La DoG actúa como un detector de "manchas" (blobs). Una vez que los puntos potenciales son localizados, se utiliza una serie de Taylor para refinar la localización de los puntos.

Una vez localizados los puntos, se calcula la magnitud del gradiente y su orientación en una vecindad de 16x16 con respecto al punto detectado. Esta vecindad es divida en 16 sub bloques de 4x4, para cada sub bloque se construye un histograma de 8 valores, así se tendrán 128 valores descriptores por cada punto detectado.

2.3 Speed Up Robust Features (SURF)

Una de las principales ventajas del algoritmo SURF, es su capacidad de calcular descriptores distintivos de forma rápida. En adición a esto, los descriptores SURF son invariantes a las transformaciones más comunes de imágenes, como lo son la rotación, los cambios de escala, los de iluminación y los pequeños cambios del punto de vista.

2.3.1 Localización de puntos de Interés

El detector SURF se basa en la matriz Hessiana (H. Bay. 2006). Dado un punto $X=(x,y)$ en una imagen I, la matriz Hessiana $H(X,\sigma)$ a una escala σ se define de la siguiente forma:

$$H(X,\sigma)=\begin{bmatrix} L_{xx}(X,\sigma) & L_{xy}(X,\sigma) \\ L_{xy}(X,\sigma) & L_{yy}(X,\sigma) \end{bmatrix} \qquad (2)$$

donde $L_{xx}(X,\sigma)$, es la convolución de la derivada de segundo orden de la gaussiana $\frac{\partial^2}{\partial x^2}g(\sigma)$ con la imagen I en el punto X, y similarmente para $L_{xy}(X,\sigma)$ y $L_{yy}(X,\sigma)$. En contraste con el algoritmo SIFT, el cual aproxima el Laplaciano del Gaussiano (LoG) con una Diferencia de Gaussianas (DoG). SURF aproxima la derivada de segundo orden con cajas de filtros como se muestra en la Fig. 1. Esto puede ser calculado de forma rápida a través de imágenes integrales. La escala y la locación de los puntos de interés son seleccionados apoyándose en el determinante de la matriz Hessiana (H. Bay, 2006). Los puntos se localizan en espacio y escala en la imagen, aplicando una supresión no máxima en una vecindad de $3\text{x}3\text{x}3$.

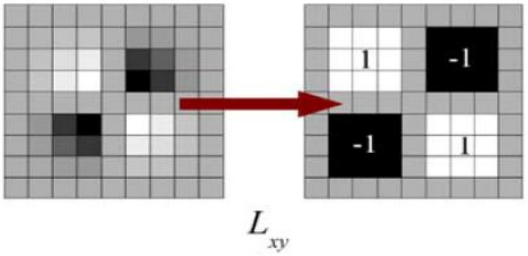


Fig. 1. Derivada parcial de segundo orden, y su correspondiente aproximación en filtro de matriz.

2.3.2 Descriptor de los puntos de interés

Una vez que se ha detectado un punto de interés, se construye una distribución de primer orden del wavelet de Haar (H. Bay, 2006) en dirección x y y de forma circular, después se construye una región cuadrada dentro de la región circular alineada con la orientación seleccionada y se extrae el descriptor SURF de esta región (H. Bay, 2006). Como lo ejemplifica la Fig. 2:

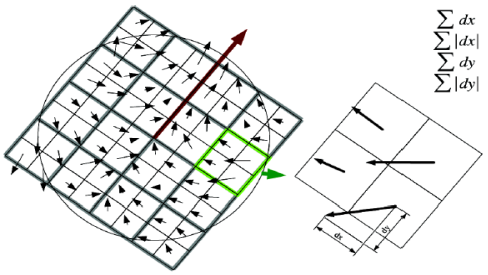


Fig. 2. Ejemplo de construcción de descriptor en base a wavelet de Haar.

Donde dx es la respuesta del wavelet de Haar en dirección horizontal para cada subregión, dy es la respuesta del wavelet de Haar en dirección vertical para cada subregión. $\sum dx, \sum |dx|, \sum dy, \sum |dy|$ representan la sumatoria de las respuestas para cada sub región.

2.4 Redes Neuronales como clasificadores y clasificadores estadísticos.

En las ultimas decadas, el uso de las redes neuronales como herramientas de clasificación han dado pie a la generación de diversas clases de redes neuronales; desde las redes neuronales de primera generación como lo es la Red Neuronal de Perceptrones Multicapa, las Redes Neuronales de Base Radial, hasta llegar a las Máquinas de Vector Soporte. En el presente artículo, se comparan los resultados de la clasificación de este tipo de redes neuronales contra otro tipo de clasificadores como lo son los clasificadores de tipo estadístico; ver sección 4.

Dentro de los clasificadores de tipo estadístico/probabilístico, encontramos aquellos que se basan en la disperción natural de las muestras a analizar; ejemplo de estos, es el clasificador ingenuo de Bayes y algunas variaciones del clasificador antes mencionado como la Red Bayessiana, el cual es un modelo gráfico probabilístico; el cual representa al conjunto de variables aleatorias y sus dependencias condicionales (I. Rish, 2001).

Dentro de los algoritmos de clasificación podemos mencionar el algoritmo K vecinos cercanos (Pádraig et al., 2007), el cual dado un número K, determina los K elementos mas representativos del conjunto, de tal forma que la clasificación se realiza sometiendo el vector de entrada a un proceso de agrupacion o clustering y los K' elementos resultantes, serán comparados contra los K x| elementos representantes. Mencionado lo anterior, en el presente trabajo se presenta un estudio comparativo de clasificación entre las redes neuronales MLP, RBNN, SVM con diferentes tipos de núcleo y los clasificadores estadísticos como lo son ingenuo Bayes, red Bayessiana y KNN.

3. VEHÍCULOS AEREOS NO TRIPULADOS

En los últimos años, el auge de los vehículos aéreos no tripulados (UAV por sus siglas en inglés) y en específico de los Quadrotores, aunado a los avances tecnológicos han empujado los límites de estos artefactos. La investigación en esta área es muy amplia y diversa. Una de las principales líneas de investigación, es la de la vigilancia aérea; que se lleva a cabo dotando al UAV con una cámara, de tal forma que mediante algoritmos de visión por computadora, se pueda analizar automáticamente una escena en particular y determinar así situaciones de peligro y poder tomar las acciones pertinentes. El trabajo que se presenta en este artículo, es un primer paso para lograr este objetivo.

3.1 Ar. Drone Parrot

El AR. Drone es un Quadrotor UAV que recibe comandos mediante una conexión inalámbrica. Este puede realizar acciones pre programadas, como lo es el despegar “Take-Off”, aterrizar “Land”, o permanecer en la misma posición en el espacio de vuelo “Hover”. Puede ser utilizado tanto en interiores como en exteriores, con y sin estructura de protección “Indoor Hull”. El AR. Drone utiliza una batería de litio recargable, es capaz de enviar información en tiempo real acerca de su estado actual, de su posición y su orientación; también puede transmitir video en tiempo real de una de las cámaras seleccionadas. El AR. Drone posee un sistema operativo Linux embebido que puede ser accedido mediante el protocolo Telnet.

3.2 Características y Componentes

El AR. Drone posee varios dispositivos de censado del medio ambiente, los cuales le permiten obtener información de la altura y posición, los cuales le ayudan a tener una mejor estabilidad en el aire. El AR. Drone posee varios sensores. Estos sensores de movimiento proveen a un PID integrado, de las medidas de deslice (yaw), de inclinación (pitch) y de empuje (roll). Estas medidas son utilizadas para la estabilización automática del Quadrotor. Un sensor de telemetría ultrasónico provee de información sobre la altitud, que a su vez es utilizada para la estabilización de la altitud y control asistido de velocidad vertical. El AR. Drone posee dos cámaras: una cámara frontal en posición horizontal, y una cámara en posición vertical con vista hacia el suelo. La cámara vertical es utilizada por el Drone para medir la velocidad relativa al suelo, y también es utilizada para realizar flotamiento automático. El AR. Drone también es capaz de enviar video codificado de cualquiera de las 2 cámaras en tiempo real. En nuestro trabajo, se utiliza principalmente la cámara vertical, que aunque se obtiene una imagen de menor calidad, la finalidad es la de poder monitorear las posibles situaciones en suelo.

4. PROCEDIMIENTO

El algoritmo propuesto en este trabajo, es un algoritmo que conjunta varias técnicas tanto de procesamiento de imágenes, como de algoritmos de clasificación y de técnicas de programación para lograr el máximo rendimiento en la clasificación en tiempo real. El diagrama general de este algoritmo se muestra en la Fig. 3:

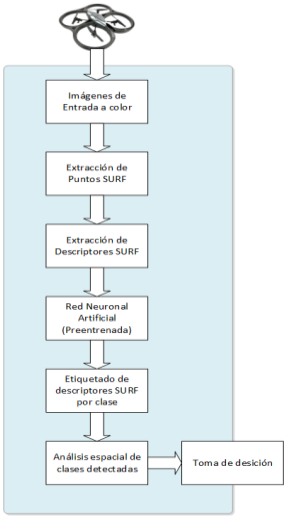
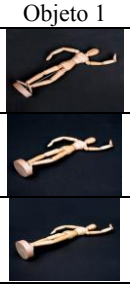
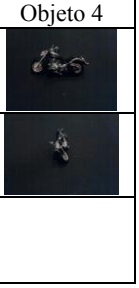


Fig. 3. Diagrama de bloques del algoritmo propuesto.

4.1 Banco de Imágenes

Claro está que en cualquier proceso de reconocimiento y clasificación de objetos, se requiere de un banco de imágenes de entrenamiento, nuestro banco de imágenes, está compuesto de las siguientes:

Tabla 1. Conjunto de objetos a procesar

Objeto 1	Objeto 2	Objeto 3	Objeto 4
			

4.2 Extracción de puntos y descriptores SURF.

Una de las principales ventajas del algoritmo SURF, es la capacidad de calcular e identificar de forma rápida los puntos característicos y sus descriptores para una imagen dada. Las imágenes que se obtienen son de tamaño de 320x240 pixeles, cuyo reducido tamaño, facilita el manejo de las mismas en cuanto a administración de memoria y tiempo de procesamiento. Una vez que se obtiene una imagen a color proveniente del Ar. Drone, a una razón de 10 imágenes por segundo (fps), se realiza la extracción de los puntos característicos de dicha imagen, el cálculo de los puntos característicos, va de la mano con el cálculo de los descriptores SURF por cada punto detectado.

Con base al conjunto de objetos presentado en la sección anterior, se obtiene el siguiente conjunto de rasgos descriptores de muestras de entrenamiento, es decir, para el Objeto 1 se tienen 900 rasgos para el entrenamiento y cada rasgo posee 64 descriptores, de igual forma para el conjunto de prueba del Objeto 1, se tienen 230 rasgos.

Tabla 2. Número de muestras por objeto.

	Objeto 1	Objeto 2	Objeto 3	Objeto 4
Entrenamiento	900	845	700	500
Prueba	230	200	150	120

4.3 RNA Entrenamiento y clasificación.

Como se mencionó en la sección 2.4, en el presente trabajo se utilizan los descriptores SURF obtenidos en la sección anterior, para identificar mediante un estudio comparativo la mejor alternativa de clasificación, la cual se selecciona de un conjunto de diferentes redes neuronales y de clasificadores de tipo estadístico. Para esto, se entrenaron las redes neuronales y los clasificadores estadísticos para un primer conjunto reducido de datos, obteniendo los siguientes porcentajes en cuanto a clasificación:

Tabla 3. Porcentajes de clasificación de diferentes clasificadores

MLP	RBNN	SVM BR	SVM Lineal	SVM Polin.	Ingenuo Bayes	KNN	Red Bayessiana
97.79	97.18	50.70	96.71	49.76	89.67	83.56	97.65

Como se observa en la tabla anterior, la red neuronal MLP y el clasificador red Bayessiana dan un buen porcentaje de clasificación (mayor al 90%), sin embargo, estos porcentajes de clasificación se obtuvieron con un conjunto de rasgos descriptores SURF reducido (de tan solo unos cientos de muestras por objeto). Por lo tanto se generó un conjunto mayor de muestras para cada objeto, entrenando una red MLP, esto nos dio el mejor desempeño en nuestro ejemplo comparativo anterior.

Tabla 4. Porcentajes de clasificación MLP para el conjunto de objetos.

Objeto	Entren.	Prueba	Clasificación entrenamiento	Clasificación Real
Objeto 1	11908	1117	99.953%	95.603%
Objeto 2	10908	738	94.248%	86.177%
Objeto 3	13077	1132	98.654%	95.060%
Objeto 4	4599	761	97.824%	96.214%

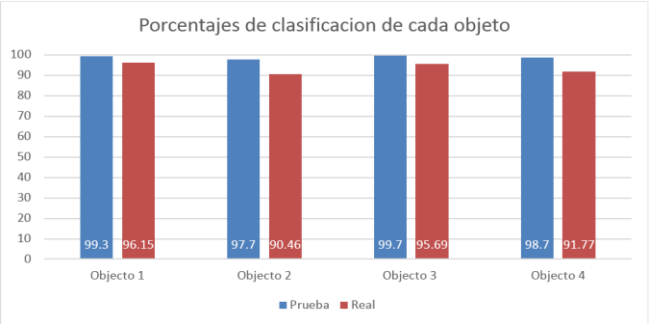


Fig. 4. Gráfica comparativa entre porcentajes de clasificación

El porcentaje de clasificación real se obtuvo sometiendo a la red neuronal entrenada a un conjunto de descriptores SURF extraídos de 100 imágenes por objeto y contabilizando los descriptores mal clasificados.

4.4 Análisis espacial de las clases detectadas.

Una vez que los descriptores SURF han sido clasificados, se observó que existen como en toda red neuronal, puntos mal clasificados. Realizando un análisis de los mismos, se observó que estos se pueden reducir en gran cantidad mediante un análisis espacial de los datos, es decir, calculando la desviación estándar y eliminando aquellos que sobrepasen la desviación estándar más un umbral, dicho umbral se calcula de forma empírica en base a la forma de cada objeto a clasificar.

Una vez filtrados los puntos, se calcula la distancia euclidiana entre los puntos de las clases detectadas, esto para determinar

la distancia entre objetos, si la distancia es pequeña se puede tipificar la escena, de lo contrario, se continúan analizando a las siguientes imágenes del video.

4.5 Ruta de vuelo y análisis en línea.

Como primer escenario, al cual llamaremos E1, se plantea simplemente el Objeto Fondo el cual es el escenario más simple. En este escenario de pruebas el algoritmo de identificación no deberá de reportar la detección de un objeto. Como segundo escenario, al cual llamaremos E2, se plantea que el Objeto 1 se presente junto con el Objeto Fondo. Como tercer escenario, al cual llamaremos E3, se plantea que el Objeto 3 se presente junto con el Objeto Fondo. Como cuarto escenario, al cual llamaremos E4, se plantea que el Objeto 2 y Objeto 4 se presente junto con el Objeto Fondo. Lo anterior se ejemplifica en la siguiente figura:

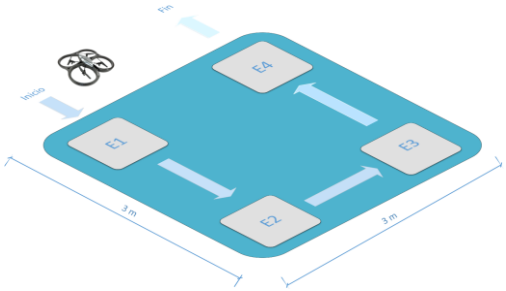


Fig. 5. Ruta de vuelo planteada.

El ambiente de pruebas con el que se cuenta es un cuarto de aproximadamente 5x5m sin techo, el cual fue elegido de esta forma para probar la robustez de los algoritmos seleccionados en cuanto al ruido producido por variaciones de luz y cambios del plano focal, es decir, las pruebas finales de clasificación se realizaron a luz de día, con la única condición de que no incidiera luz directamente sobre los escenarios de prueba (entre las 11 am y las 3 pm), ya que a estos horarios se genera una sombra producida por el Quadrotor, lo cual dificulta el análisis de las imágenes. La ruta de vuelo comprende en un área de 3x3m como se muestra en la Fig. 5.

4.6 Consideraciones de velocidad de vuelo y ruido inducido por la plataforma.

Para poder llevar a cabo la ruta vuelo sobre los escenarios antes mencionados, se tuvo que configurar la velocidad máxima angular (VA) de los motores, así como el ángulo de inclinación con respecto al eje horizontal θ (Carlos Eduardo, November 2011), esto con el fin de que el desplazamiento en el aire del Ar. Drone sea lento y lo más estable posible. Se proponen los siguientes valores en base a las observaciones realizadas en los experimentos:

$$VA = w_{\max} * 0.5 = 1000 mm/s \quad (3)$$

$$\theta = \theta_{\max} * 0.1 = \pm 3^{\circ} \quad (4)$$

Donde $w_{\max}$ es la velocidad angular máxima de los motores, que según las hojas de especificación del Ar. Drone es de 2000 mm/s. Y $\theta_{\max}$ es el ángulo máximo de inclinación el cual tiene un valor de $\pm 30^{\circ}$.

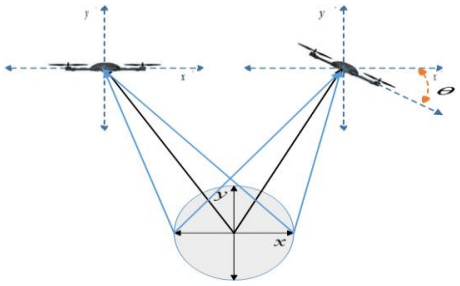


Fig. 6. Ejemplo de variación del ángulo de inclinación para lograr un desplazamiento del Quadrotor.

En el presente trabajo se decidió despreciar el ruido inducido por los motores, ya que en la fase de experimentación se observa que la distorsión producida en las imágenes es mínima.

5. RESULTADOS

Del análisis efectuado en los puntos 4.2 y 4.3, podemos afirmar que la clasificación de los descriptores SURF mediante redes neuronales artificiales es factible y aún más, con la misma red neuronal (en nuestro caso una MLP), pueden ser clasificados varios objetos con una confiabilidad en la clasificación mayor al 89%.Las siguientes imágenes que se muestran como prueba de la clasificación en tiempo real, fueron tomadas con el Quadrotor en operación (es decir en vuelo), a una altura aproximadamente de 1.50m de altura.

Se presenta cada uno de los objetos involucrados por separado y en conjunto, lo cual demuestra la clasificación por objeto de forma satisfactoria. La tabla 5 muestra los colores asignados a los descriptores de cada objeto, los cuales se deberán de ver reflejados en la imágenes a la hora en que se clasifican dichos descriptores.

Tabla 5. Colores asignados a los descriptores de cada objeto.

Objeto	Color asignado al descritor	Color
Objeto 1		Rojo
Objeto 2		Verde
Objeto 3		Azuk
Objeto 4		Negro
Fondo		Amarillo

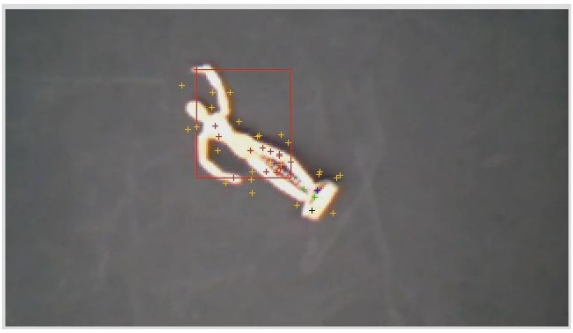


Fig. 7. Ejemplo de clasificación del escenario 2.

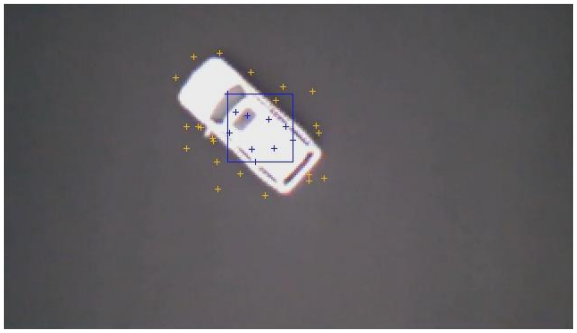


Fig. 8. Ejemplo de clasificación del escenario 3.

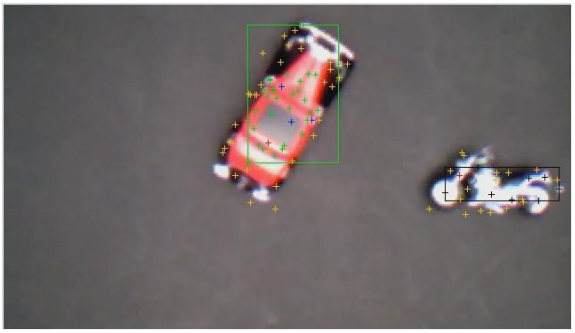


Fig. 9. Ejemplo de clasificación del escenario 4.

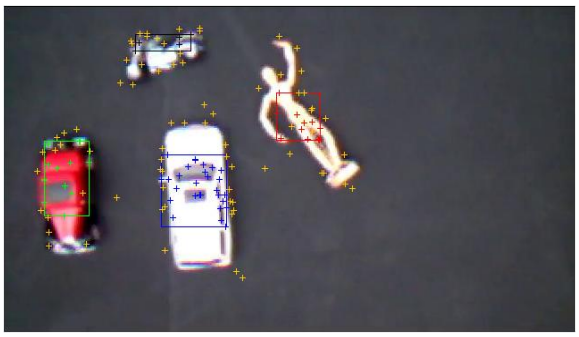


Fig. 10. Ejemplo de clasificación con varios objetos.

6. CONCLUSIONES

En el presente trabajo, se presentó la conjunción de varios algoritmos y metodologías con el fin de poder identificar y clasificar escenas simuladas de accidentes, por medio de imágenes tomadas por un Quadrotor. Las técnicas utilizadas nos presentan un primer paso para el desarrollo de sistemas más robustos y completos. Demostrando con lo anterior que en un ambiente controlado es posible tipificar una escena en tiempo real y tomar las decisiones pertinentes.

AGRADECIMIENTOS

Gerardo Hernández agradece el apoyo económico por parte de CONACyT, México. Los autores agradecen el apoyo brindado al IPN, a través de la COFAA; a la SIP-IPN a través de los proyectos: SIP 20131505, SIP 20131182, SIP 20144538, SIP 20140776 al CONACyT 155014.

REFERENCIAS

Aydin Eresen, Nevrez Imamoglu. (2012). Autonomous quadrotor flight with vision-based obstacle avoidance in virtual environment. pp. 894-905.

Carlos Eduardo Pereira. (2011). A Java Autopilot for Parrot ARDrone.

D. Lowe. (1999). Object recognition from local scale-invariant features. pp. 1150-1157.

D. Lowe. (2004). Distinctive Image Features from Scale-Invariant Keypoints. pp. 99-110.

David Kriesel, A brief introduction to Neural Networks, http://www.dkriesel.com/en/science/neural_networks.

Donghoon Kim, Rozenn Dahyot. (2006) Face Components Detection using SURF Descriptors and SVMs. pp. 51-56.

H. Bay. Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. (2006). Speeded-Up Robust Features (SURF). pp. 404-417.

I. Rish. (2001). An empirical study of the naive Bayes classifier. T.J. Watson Research Center.

Pádraig Cunningham, Sarah Jane Delany. (2007). k-Nearest Neighbour Classifiers.

Pengfei Fang, Jianjiang Lu, Yulong Tian, Zhuang Miao. (2011). An improved object tracking method in UAV videos. pp. 634-638.

Rafael C. Gonzalez, Richard E. Woods. (2002). Digital Image Processing Second Edition. pp. 349-404.

Richard Szeliski, Computer Vision, Algorithms and Applications. pp. 205-259.

U. Niethammer, S. Rothmund, U. Schwaderer, J. Zeman, M. Joswig. (2011). Open source image processing tools for low cost UAV, landslice investigations, Remote Sensing and Spatial Information Sciences.

Wulfram Gerstner, Werner M. Kistler. (2002). Spiking Neuron Models Single Neurons, Populations, Plasticity. pp. 41-75.

Xi Chao-jian,Guo San-xue. (2011). Image Target Identification of UAV Based on SIFT. pp. 3205-3209.

Yingcai Bi, Haibin Duan. (2013). Implementation of autonomous visual tracking and landing for a low-cost Quadrotor, pp. 3296-3300.