



## Monitoreo Automático de Carreteras Mediante el Uso de un Sistema de Detección, Seguimiento y Extracción de Características Básicas de Vehículos con Técnicas de Visión por Computador

Jorge E. Faytong, Giancarlo J. Moggia, Boris X. Vintimilla Ph. D  
Centro de Visión y Robótica

Facultad de Ingeniería Eléctrica y Computación  
Escuela Superior Politécnica del Litoral

Apartado 09-01-5863. Km. 30.5 vía Perimetral, Guayaquil, Ecuador

[jfaytong@fiec.espol.edu.ec](mailto:jfaytong@fiec.espol.edu.ec) [gmoggia@fiec.espol.edu.ec](mailto:gmoggia@fiec.espol.edu.ec) [boris.vintimilla@espol.edu.ec](mailto:boris.vintimilla@espol.edu.ec)

### Resumen

*En este proyecto se pretende implementar un sistema que usa técnicas de visión por computador para detectar, seguir y extraer características básicas de vehículos que se encuentran circulando en carreteras o autopistas. Para probar el sistema implementado se prevé usar como banco de prueba el uso de videos que han sido generados en vías rápidas o carreteras a través del uso de cámaras. Los resultados obtenidos con el sistema propuesto ofrecen información básica de los vehículos, tales como: dirección de su movimiento, carril en el que se desplaza, tamaño de la caja englobante que lo contiene, entre otros.*

*La información que se genera de este proyecto podría servir como base para ayudar a monitorear el flujo vial en carreteras o autopistas, lo cual posteriormente facilitará las labores de control vial de los servidores públicos y la empresa privada. No es arriesgado pensar que en un futuro cercano se podría reutilizar o adaptar el sistema para realizar seguimiento vehicular en zonas urbanas. Adicionalmente, podría también considerarse su uso total o parcial para proyectos similares en túneles, vigilancia en video, sistemas de llamadas de emergencia, sistemas de peaje, entre otras aplicaciones afines.*

**Palabras Claves:** *visión por computador, seguimiento vehicular, detección de movimiento, modelo Gaussiano, filtro de Kalman.*

### Abstract

*This project aims to implement a system to detect, track and extract basic characteristics of moving vehicles while in road or highway transit. The system will use recorded videos of road or highway transit by camera for testing. The obtained results offer basic information of the vehicles, such as motion direction, lane in which it is moving, containing bounding box's size, etc.*

*The information generated by this project could be used as a base to aid road transit monitoring in roads or highways. It is correct to think that this system could be reused or adapted to perform vehicle tracking in urban areas in a near future. Furthermore, its partial or total used could be considered for similar projects in tunnels, surveillance video systems, emergency call systems, tolls, and similar transit related applications.*

**Keywords:** *Computer vision, vehicle tracking, motion detection, Gaussian model, Kalman filtering*

Revisado OK. 08 / Febrero / 2009

Dr. Boris X. Vintimilla Burgos



## 1. Introducción

Hoy en día el tráfico vehicular es complicado en la mayoría de las áreas metropolitanas y rurales de casi todos los países en los que la congestión y el mantenimiento de las vías se han convertido en problemas cotidianos de difícil solución.

Una de las respuestas más eficientes al problema del mantenimiento de vías y la congestión radica en el uso intensivo de sistemas informáticos y de las telecomunicaciones aplicadas a la gestión del tráfico. Los denominados *Sistemas Inteligentes de Transporte (SIT)*, son un ejemplo que incluye electrónica avanzada, comunicaciones y sistemas informáticos para aumentar la eficiencia y seguridad del transporte por carretera y su mantenimiento.

El monitoreo de tránsito vehicular es un área en la que se están utilizando sistemas avanzados de visión por computador y electrónica para mejorar el control del tráfico y para realizar un mejor estudio de necesidades para los programas de manutención vial. Para obtener un monitoreo personalizado y robusto, con extracción automática de características, los sistemas de transporte inteligente poseen un sinnúmero de técnicas a su disposición.

Los métodos basados en conocimiento, véase [18, 19, 20, 22], tienen la ventaja de poseer un costo de implementación relativamente más bajo. Sin embargo, poseen la tremenda desventaja de la eliminación de regiones de interés en su etapa de detección, por lo que brindan resultados no muy concisos y específicos. Por otro lado, los sistemas que emplean técnicas de reconocimiento en estéreo, véase [18, 25, 26], ofrecen resultados sumamente concisos y exactos. Lastimosamente tienen como desventaja que los recursos computacionales demandados son sumamente altos. Finalmente se encuentran los sistemas con *detección basados en movimiento*. Estos son capaces de encontrar objetos y obstáculos móviles utilizando las técnicas de *diferenciación temporal, flujo óptico y background subtraction*, véase [13, 14, 17, 24, 25].

Existen sistemas y aplicaciones con este método en forma tanto discreta como continua. Es un hecho que las formas discretas de implementación de esta técnica son las que mejores y más eficientes resultados ofrecen. Los sistemas que utilizan el background subtraction tienen múltiples ventajas. Algunas válidas de mencionar son que ofrecen resultados excepcionales y altamente descriptivos, no introducen ruido al proceso en cada una de las ejecuciones del sistema, es también extremadamente adaptable a las

necesidades y condiciones para las cuales tengamos en mente darle uso y su carga computacional es sumamente modesta.

El sistema desarrollado en nuestro proyecto incluye la técnica de background subtraction por las ventajas mencionadas arriba y descritas en detalle en el capítulo 2. Adicional a esta técnica se implementó un algoritmo para seguir a los vehículos en movimiento, en nuestro caso una combinación de mezclas Gaussianas. Gracias a esta técnica de seguimiento de vehículos, el sistema obtuvo una mayor calidad en los resultados al ser más explícitos. El sistema es también capaz de determinar ciertas características básicas de los vehículos mediante cálculos simples pero eficaces. Nuestra investigación se vio reforzada incluso un poco más con la acertada inclusión de algoritmos adaptativos lineales de predicción del movimiento.

## 2. Trabajos previos

Usualmente un sistema de vigilancia de objetos en movimiento a través de visión por computador está formado por 3 módulos, que son: detección de objetos en movimiento, seguimiento de los objetos en movimiento y análisis del movimiento.

La detección de objetos en movimiento es el primer paso a considerar en el sistema de visión por computador. En los trabajos previos, varios métodos de detección de movimiento han sido propuestos y pueden ser divididos en tres categorías:

- Basados en el conocimiento
- Basados en estéreo
- Basados en el movimiento

Algunos ejemplos de métodos conocidos basados en el conocimiento son explicados a continuación. Bertozzi *et al* [18] propuso un método produciendo un mapa simétrico mediante la combinación entre las simetrías a nivel de grises, de ejes horizontales y verticales. El método ofrece una gran exactitud en la detección en situaciones aisladas, con secuencias de imagen sin presencia de ruido. Sin embargo, tiene defectos en casos especiales, como cuando cruzan dos carros y uno traslapa a otro en la imagen, creándose entonces falsas detecciones y cajas englobantes inadecuadas.

Los métodos basados en estéreo toman mucha ventaja del mapeo de la perspectiva inversa (IPM) para estimar las posiciones de los objetos y obstáculos en una imagen. Suhr *et al* [25] empleó un lector láser para modelar en tres dimensiones el mapa global. Se definen puntos de referencias, a partir de los cuales se aplican métodos de de-rotación y se analizan con un mosaico de 3D para estimar un parecido a una forma ya predefinida. Lastimosamente, su valor de implementación es bastante elevado. Sin embargo, su

Revisado OK. 08 / Febrero / 2009

Dr. Boris X. Vintimilla Burgos

efectividad es muy alta y es recomendable en caso de poseer un presupuesto cuantioso.

Finalmente, los métodos de detección basados en movimiento son capaces de encontrar objetos y obstáculos utilizando la técnica de flujo óptico. Esta técnica consiste en detectar el movimiento aparente de patrones de brillo en una imagen, mediante la generación de un vector de desplazamiento para cada pixel (aproximación continua). Las técnicas basadas en el flujo óptico son muy populares y efectivas para el proceso de detección. El flujo óptico es en si el patrón de movimiento aparente de los objetos, superficies y bordes en una escena visual causada por el movimiento relativo entre un observador (un ojo o cámara) y la escena.

Es necesario mencionar que además de la clasificación de las 3 categorías para detectar movimientos que fueron descritos arriba, en la bibliografía existen también otras clasificaciones para estos métodos. Se menciona la existencia de tres aproximaciones convencionales para la detección del movimiento:

- Diferenciación temporal (de dos o tres frames), véase Anderson *et al* [35], para una explicación más detallada.
- Background subtraction, Haritaoglu *et al* [37] y Wren *et al* [14], exponen este método de forma interesante.
- Flujo óptico, detalles en Barron *et al* [36].

Las técnicas de diferenciación temporal son muy adaptables a ambientes dinámicos. Sin embargo, realizan generalmente un desempeño pobre en la extracción de todos los píxeles con características relevantes. El método de *background subtraction*, o también conocido como *substracción de fondo*, es una técnica muy usada y sencilla de implementar. Esta técnica provee la más extensa colección de datos de los píxeles en movimiento con características relevantes, pero es sensible a los cambios dinámicos de escena, generalmente causados por variaciones en la iluminación y factores externos.

Shih y Chang *et al* [29] exponen un método para la detección de objetos en fondos no estacionarios. Los movimientos concurrentes de los píxeles del foreground<sup>1</sup> y background hacen que el

<sup>1</sup> El *foreground* representa a la imagen de primer plano y contiene a los objetos que están en movimiento. Mientras que, el *background* representa a la imagen de fondo o segundo plano, la cual contiene a todos los objetos estacionarios (que no se mueven).

mantenimiento plausible de un modelo de fondo para background subtraction sea sumamente complicado.

Fujiyoshi, Lipton y Kanade *et al* [17] hacen hincapié en el seguimiento de objetos, en su caso de humanos, y que contribuyen a la generación de modelos de patrones de movimiento de diferentes objetos, información sumamente importante para una posterior clasificación acertada con procesamiento no tan extenso, así como también para un posterior análisis del movimiento. Es importante mencionar que para movimientos de objetos más complejos es sumamente útil la técnica propuesta por ellos, sin embargo, al nosotros manejar bloques con movimiento en 2D, no es necesario utilizar los métodos y aproximaciones por ellos implementados.

En los métodos de detección de movimiento debemos considerar que los objetos de la imagen de background pueden permanecer estacionarios constantemente o pueden también variar cada cierto tiempo, para luego permanecer estacionarios. Es debido a estos casos que el background debe ser actualizado cada cierto tiempo. En este sentido, el *modelo adaptativo Gaussiano* de actualización del background consiste en un algoritmo de actualización inteligente de la imagen utilizada como fondo en el seguimiento de un vehículo en un video. Es una herramienta que utiliza una mezcla de variables con distribución Gaussiana para la aplicación del filtro de actualización. A continuación describiremos el algoritmo del modelo Gaussiano simple (SGM) propuesto por Wren *et al* [14]. En este método, la intensidad y el color de cada pixel es representado por un vector  $[Y,U,V]^T$ . Se asume que solo existen cambios lentos en las escenas. La media y la covarianza de cada pixel A puede ser recursivamente actualizado de esta forma:

$$\begin{aligned} \mu^{\square}(\varphi) &= (1-\alpha)\mu^{\square-1}(\varphi) + \alpha I^{\square}(\varphi) \\ U^{\square}(\varphi) &= (1-\alpha)U^{\square-1}(\varphi) + \alpha v(\varphi)v(\varphi)^T \end{aligned}$$

Donde  $I^{\square}(\square)$  es el pixel del frame actual en un espacio de color Y-UV y  $\alpha$  es la tasa de aprendizaje.

$$l(\varphi) = -\frac{1}{2}v(\varphi)^T(U^{\square})^{-1}v(\varphi) - \frac{1}{2}\ln|U^{\square}| - \frac{3}{2}\ln 2\pi$$

Después de una adaptación de fondo, computamos todas las posiciones de las imágenes  $\square$  y la diferencia entre la imagen actual y el fondo. Este valor da origen a una clasificación individual de píxeles como fondo o foreground. Un pixel  $\alpha$  es clasificado como foreground si  $l(\varphi) < n_c$ . En otro caso es parte del

fondo. Después de esto se pasa a detectar los puntos del foreground.

Por otro lado, el modelo Gaussiano múltiple utiliza combinaciones de modelos Gaussianos de fondo, los cuales han sido una opción muy popular para modelar fondos complejos y variantes en el tiempo. En este algoritmo el procesamiento de píxeles es considerado una serie de tiempo de vectores para imágenes de color. La historia de un pixel  $\alpha$  en particular está dado por:

$$\{X_1, \dots, X_t\} = \{I(\alpha, i) : 1 \leq i \leq t\}$$

donde  $I$  es la secuencia de imágenes. El algoritmo modela la historia reciente de cada pixel como una mezcla de  $K$  distribuciones Gaussianas. De la misma forma, la probabilidad de observar el valor actual del pixel es:

$$P(X_t) = \sum_{i=1}^K w_{i,t} \cdot \eta(X_t, \mu_{i,t}, U_{i,t}) \dots$$

Donde  $K$  es el número de distribuciones,  $w_{i,t}$  es el peso estimado del  $i$ -ésimo Gaussiano en la mezcla en un tiempo  $t$  y  $U_{i,t}$  y  $\mu_{i,t}$  son la media y la covarianza de la matriz del  $i$ -ésimo Gaussiano en un tiempo  $t$  y  $\eta$  la función de densidad de la probabilidad Gaussiana.

$$\eta(X_t, \mu, U) = \frac{1}{(2\pi)^{\frac{n}{2}} |U|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t - \mu)^T U^{-1} (X_t - \mu)}$$

El algoritmo asume que el rojo, azul y verde son canales independientes y que cada pixel no es estacionario. Estos supuestos dan como resultado  $K$  medias de aproximación en tiempo real para el algoritmo de mezcla. En la aproximación en tiempo real, cada nuevo pixel  $X_t$  es chequeado con un  $K$  ya existente de la distribución Gaussiana. Un valor de respuesta es encontrado si el valor del pixel está dentro de la desviación estándar de una distribución.

Es en sí, un threshold bastante efectivo para cada pixel y puede ser utilizado para modelar regiones que presentan cambios periódicos en las condiciones de alumbrado. Si el pixel actual no coincide con las distribuciones, se reemplaza a la distribución menos probable con este valor, agregándole una alta varianza y un peso de prioridad bajo. Este algoritmo es sumamente eficiente y a la vez adaptable a una gran variedad de situaciones. La sensibilidad a los cambios de brillo es aún la desventaja más prominente. Sin embargo, gracias al procesamiento interno del algoritmo, aparece en un menor grado que en otras técnicas similares.

Los filtros de predicción de estados posteriores de un objeto afectado por un sistema han jugado un rol importantísimo en el avance de los productos y servicios que ahora disfrutamos en muchos aspectos de nuestras vidas.

El observador de Luenberger, mencionado por Bundhwar *et al* [5], “clona” el sistema evaluado, creando otro sistema, el cual es conocido, e introduciendo las mismas entradas a ambos. A partir de la salida generada por el sistema en evaluación, el algoritmo modifica la salida obtenida por el clon, adaptando este último resultado al primero, lo cual generalmente es la salida del clon más la diferencia entre la salida de los dos sistemas multiplicada por una constante. El objetivo final, es poder modelar un sistema desconocido a partir de uno conocido y así definir un comportamiento esperado para el sistema en estudio.

Un algoritmo de predicción basado en la diferencia de las posiciones de dos puntos previamente obtenidos de manera satisfactoria es el del desplazamiento previo, propuesto por Efron *et al* [48]. Consiste en realizar mediciones constantes de la variación entre los estados previos de un objeto sujeto a un sistema no conocido, para predecir un posible siguiente estado. Posee una ventaja muy apreciable al no necesitar que el sistema que está afectando al objeto sea definido determinístico o abstractamente.

El filtro Kalman, propuesto por Kalman *et al* [8], es un método de estimación cuyos parámetros se corrigen en cada iteración dependiendo del error de predicción que se haya cometido en la iteración anterior. Es un estimador lineal y óptimo desde el punto de vista de mínimos cuadrados, que ha ganado aceptación en el análisis de series de tiempo.

Los siguientes apartados están organizados como sigue. En la sección 3 se describen los módulos de generación de imágenes a procesar y de detección de vehículos en movimiento mediante la técnica de background subtraction. La sección 4 describe cómo ha sido implementado el módulo de seguimiento de los vehículos previamente detectados con movimiento así como también detalla cómo han sido extraídos las características básicas de dichos vehículos mediante técnicas de extracción de blobs, caja englobante y determinación de centroide. Finalmente, los resultados y conclusiones son presentados en la sección 5.

### 3. Generación de imágenes a procesar y detección de movimiento

Dado que nuestro sistema no ha sido desarrollado para trabajar en tiempo real de manera óptima, usaremos como datos de entrada archivos de video que han sido previamente obtenidos desde una

Revisado OK. 08 / Febrero / 2009

Dr. Boris X. Vintimilla Burgos

autopista o carretera. Nuestro sistema no contempla el montaje y calibración de las cámaras responsables de obtener los videos viales desde las autopistas o carreteras. Nuestro sistema utiliza una interfaz gráfica que permite cargar un archivo de video ya existente.

Para el desarrollo del sistema nos valemos de la librería gráfica MFC del framework de desarrollo de Windows, *WinAPI*. Toda la interfaz gráfica, denominada *ImageCapture*, es desarrollada con funciones y métodos de MFC, con los que tenemos la posibilidad de crear ventanas, en las cuáles podemos visualizar los videos y el procesamiento realizado por diferentes algoritmos implementados.

Para realizar la captura en sí, se necesitan funciones de acceso a la cámara y de obtención de frames. Para esto utilizamos la librería de visión por computadora de código abierto, *OpenCV*.

*OpenCV* (*Open source computer vision*) es una biblioteca libre de visión artificial por computador originalmente desarrollada por Intel y publicada bajo licencia BSD. Este marco de trabajo facilita en gran manera el aprendizaje e implementación de distintas técnicas de visión por computador, tanto a nivel docente como investigador. La biblioteca es multiplataforma, y puede ser usada en Mac OS X, Windows y Linux. La *OpenCV* es usada ampliamente por los desarrolladores de aplicaciones de visión por computador y debido a esto tiene una comunidad numerosa de adeptos e investigadores.

Una vez que el archivo de video ha sido cargado a nuestro sistema, el siguiente paso consiste en capturar cada una de las imágenes o frames que constituyen dicho video. La figura 1 muestra un ejemplo de imágenes capturadas desde un video.



**Figura 1. Imágenes adquiridas en un video: (izq. a der.) frame 15, frame 30, frame 45 y frame 60.**

El módulo de detección de movimiento tiene como propósito único y exclusivo detectar de la mejor manera los objetos en movimiento dentro de una secuencia de imágenes. El sistema desarrollado maneja la detección en dos partes:

- Generación de la imagen de fondo o background
- Generación de la imagen de movimiento o Foreground

Es necesario manejar una imagen de fondo capaz de adaptarse a muchos factores, siendo los más importantes:

- Cambios en la iluminación: tanto gradual como repentina (nubes)
- Pequeños cambios en el movimiento
- Pequeñas oscilaciones de la cámara
- Cambios en la geometría del fondo fijo: como carros parqueados.
- Objetos con alta frecuencia de aparición en el fondo fijo: como ramas de árboles, olas y cosas similares.

Nuestro sistema captura y actualiza el fondo con cada frame procesado sin excepción. La adaptación a los cambios se garantiza al saber con seguridad que el fondo se adaptará con cada nuevo frame. Estando entonces la imagen de referencia en constante adaptación, se asegura un fondo o background preciso durante todo el tiempo de ejecución. Una vez obtenido el fondo se procede a generar la imagen de movimiento o foreground.

La generación de la imagen de movimiento es esencial en nuestro sistema ya que define cuáles regiones se encuentran en movimiento. Este procedimiento es muy similar al de la obtención del fondo o background. De manera equivalente, se tiene que realizar un proceso de manera ordenada y debe ser ejecutado en el procesamiento de todos y cada uno de los frames. La figura 2 ilustra las imágenes de background y fondo generadas a partir de una imagen de entrada.



**Figura 2. Imágenes de: (izq.) background, y (der.) foreground obtenida desde una imagen de entrada (centro).**

Como se puede apreciar en la Figura 2 (der.), imagen resultante en la detección de movimiento, al dividir las zonas de movimiento con las zonas inmóviles se ve afectada por la generación de regiones o píxeles con ruido. En muchas ocasiones se obtienen pequeñas zonas de movimiento falsas, es decir, donde realmente no hay movimiento, o el movimiento es mínimo, o no involucra a un vehículo, de manera que se lo considera despreciable. Para solucionar este inconveniente recurrimos a la eliminación de ruido mediante operaciones de filtrado morfológico.

Revisado OK. 08 / Febrero / 2009

Dr. Boris X. Vintimilla Burgos

## 4. Seguimiento y extracción de características básicas de vehículos en movimiento

En nuestro sistema, el seguimiento de vehículos está compuesto por los submódulos de:

- Segmentación
- Caja englobante

La segmentación se encarga de darle sentido a la información resultante del proceso de detección. Para el procesamiento de estos datos, el módulo de segmentación se basa en la utilización de una librería para la extracción de *blobs*, desarrollada para OpenCV. Un *blob* es una estructura compuesta por un conjunto de píxeles adyacentes y sus atributos. Este conjunto de píxeles han sido agrupados tomando en cuenta que ellos cumplen con ciertos criterios o parámetros de clasificación y es tomado como un objeto, no como simples píxeles separados y sin sentido. La librería para la extracción de *blobs* tiene el nombre de *cvblobslib* y nos proporciona funciones muy sencillas que nos facilitan de gran manera la segmentación de vehículos.

*Cvblobslib* es una librería estática desarrollada en C++, capaz de ejecutar etiquetación de componentes a imágenes binarias. La *cvblobslib* también provee de funciones para manipular, filtrar y extraer resultados de los *blobs* generados. Un *blob* generado con el uso de esta función es mostrado en la figura 3.

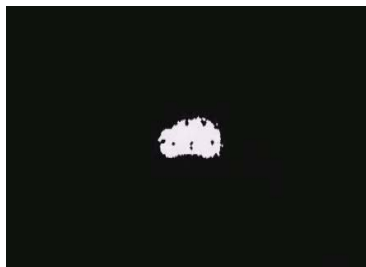


Figura 3. Blob segmentado

Por otro lado, el módulo de caja englobante se encarga de encerrar a los *blobs* encontrados para hacer posible la adquisición de datos de entrada necesarios para los componentes de extracción de características básicas y predicción de estados futuros. Es importante mencionar que este módulo es el encargado de la obtención del centroide de cada *blob*. El centroide es el centro de la caja englobante del *blob*. Es en sí la estructura de datos utilizada como entrada principal en los componentes antes mencionados, por lo que su producción es vital.

Adicional a los 2 módulos anteriores, existe el módulo de extracción de características básicas. Este módulo se desarrolló con el fin de brindarle al usuario una información más completa, describiendo de manera más detallada el movimiento realizado por el vehículo. De esta forma, es posible extraer características tales como: dirección de su movimiento, carril en el que se desplaza, tamaño de la caja englobante que lo contiene, entre otros. Para esta sección también se utiliza a la *cvblobslib*.

Reforzamos el sistema agregándole un módulo de predicción de estados futuros, con lo cual el modelo obtiene mayor robustez al ser capaz de entregar una respuesta incluso cuando los resultados de la detección contienen demasiado ruido, son incongruentes o en el peor de los casos, inexistentes. Este módulo utiliza exclusivamente funciones de OpenCV para su procesamiento. La figura 4 ilustra los resultados obtenidos de esta sección.



Figura 4. Vehículo dentro de la caja englobante con la predicción de la siguiente posición (punto color verde), punto azul representa al centroide.

## 5. Resultados y conclusiones

Las pruebas presentadas en este documento fueron realizadas con videos obtenidos en el área de ingenierías de la ESPOL. La cámara fue colocada en el techo de la biblioteca central de ingenierías, de donde se obtuvieron porciones largas de video con una cámara digital. Estas capturas fueron mucho más nítidas y menos ruidosas que las anteriores. Se pudo experimentar con mayor capacidad de análisis y mayores conclusiones fueron sacadas.

A continuación una tabla con información relevante para la óptima configuración del sistema:

Tabla 1 Configuración óptima recomendada para grabar el video a analizar

|                      |           |
|----------------------|-----------|
| Altura cámara        | 14,7 m.   |
| Distancia de enfoque | 10,8 m    |
| Ángulo               | -42°      |
| Resolución cámara    | 640 x 480 |

Revisado OK. 08 / Febrero / 2009

Dr. Boris X. Vintimilla Burgos

Las pruebas fueron realizadas en tres horarios:

- 08h30-09h05
- 12h30-13h40
- 18h30-19h00

Pudimos observar que la detección más cercana al resultado real ocurría en el horario del medio día (200 de 201 detectados), por lo que a esa hora la visibilidad fue la mejor de los tres casos. Como se aprecia también, el peor desempeño del sistema ocurrió en el horario de 18h30 a 19h00. Esto se debe a la mala iluminación y por ende la enorme cantidad de ruido presente y falta de información de calidad en los datos de entrada.



**Figura 5. Resultado final**

Se pudo determinar casos especiales donde el sistema es incapaz de detectar un vehículo si es demasiado grande, como el caso de un autobús, ya que en este caso el automotor cubre el área de detección de manera absoluta, por lo que el sistema se ve imposibilitado de encerrarlo en una caja englobante.

A su vez, se pudo constatar que el sistema efectivamente detecta exclusivamente vehículos y la presencia de ruido externo, como el paso de personas o animales es obviado con gran eficacia.

En los resultados mostrados se pudo apreciar que el estado pronosticado por el filtro de Kalman coincide en muchos casos con la medida real obtenida. La técnica del desplazamiento previo provee también una aproximación muy buena, pero falla en ciertos puntos, desviándose de la trayectoria real.

Por otro lado, la alta independencia y robustez del filtro de Kalman, comparado con el filtro del desplazamiento previo le otorgan ventaja a la hora de seleccionar un filtro predictor adaptativo.

Muchos algoritmos de extracción de background que no han sido discutidos aquí asumen que el background no varía y entonces puede ser capturado previamente. Esto limita enormemente su utilidad en la implementación práctica. Muy pocos documentos describen y discuten su algoritmo con el suficiente detalle para que sea fácil su implementación.

Aunque el sistema es bastante económico en lo que a recursos computacionales se refiere, es necesario que corra bajo ciertos parámetros estrictos de exclusividad de uso sobre un procesador de alto rendimiento, ya que se trata de cálculos en tiempo real. Sin embargo, esto no debería ser un gran inconveniente, ya que las compañías encargadas de construcciones viales y la empresa pública manejan cifras mucho más altas para este tipo de estudios.

Por otro lado, los resultados obtenidos muestran una alta aproximación a lo real, con lo que el sistema queda justificado. Es decir, se ha podido obtener un sistema robusto y efectivo, que podría brindar asistencia para el control de carreteras rurales o estudio del comportamiento vehicular de las mismas.

## 7. Referencias

1. N. Siebel, “**The Reading people tracker**”, **proyecto en ejecución, para más información visite:**  
[http://www.siebelresearch.de/people\\_tracking/reading\\_people\\_tracker/](http://www.siebelresearch.de/people_tracking/reading_people_tracker/)
2. D.Hall, J. Nascimento, P. Ribeiro, E. Andrade, P. Moreno, S. Pesnel, T. List, R. Emonet, R.B. Fisher, J. Santos Victor y J.L. Crowley, “**Comparison of target detection algorithms using adaptive background models**” Proceedings on 2nd Joint IEEE International Workshop on Visual Surveillance and tracking, 2005, pg. 113-120.
3. M. Piccardi, “**Background subtraction techniques: a review**” **Proceedings of 2004 Systems, man and cybernetics, volume 4**, 2004, pg. 3099-3104.
4. A. Efros, “**Rendering and Image Processing**” tomado de la biblioteca de Ciencias de la Computacion, de la facultad de Computacion de la universidad de Carnegie Mellon, última versión en <http://www.cs.cmu.edu/~efros/>
5. D. Simon, “**Kalman filtering**” página web de investigacion Scribd, última version en <http://www.scribd.com>
6. D. Bundhwar, S. Sandhu, A. Karna “**Target tracking using Kalman filter**” página web de investigacion Scribd, última version en <http://www.scribd.com>
7. D. Peña, “**Sobre la robustificación interna del algoritmo de Plackett-Kalman para la estimación recursiva del modelo de**



- regresión lineal**” Trabajos de estadística e investigación operativa, Springer Berlin / Heidelberg, Berlín, Alemania, Julio 1985, pg. 93-106.
8. R. Kalman, “**A New Approach to Linear Filtering and Prediction Problems**” Transactions of the ASME – Journal of Basic Engineering, revista. 82, 1960, pg. 35-45
  9. J. Perrin, S. Volino, “**Gaussian model for localized translational motion: Application to incoherent neutron scattering**” The European Physical Journal: Special topics, Springer Berlin / Heidelberg, Berlín, Alemania, Julio 2007, pg. 57-60.
  10. A. Calway, R. Wilson, “**Multiresolution Gaussian mixture models for visual motion estimation**” Proceedings 2001 International Conference on Image Processing, volumen 2, 2001, pg. 921-924.
  11. D.S.G. Pollock, “**The Kalman Filter**” Computational Statistics and Data analysis, volumen 50, 1 May 2006, pg. 2137-2145.
  12. W. Zhang, X. Fang, W. Lin “**Moving vehicles segmentation based on Gaussian motion mode**” Visual Communications and Image Processing 2005, volumen 5960, 2005, pg. 141-148.
  13. G. Halevy, D. Weinshall “**Motion of disturbances: detection and tracking of multibody non-rigid motion**” 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volumen 17, 1997, pg.897-902.
  14. C. Wren “**Pfinder: Real time tracking of the human body**” IEEE Transactions on Pattern Analysis and Machine Intelligence, volumen 19, 1997, pg. 780-785.
  15. Y. Huang “**A vision based vehicle to vehicle detection and tracking system**” ICPR 2006. 18th International Conference on Pattern Recognition, volumen 2, 2006, pg. 1070-1073.
  16. N. Kanhere, S. Pundlik, S. Birchfield “**Vehicle segmentation and tracking from a low able off axis camera**” Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volumen 2, 2005, pg. 1152-1157.
  17. H. Fujiyoshi, A. Lipton, T. Kanade “**Real time human motion analysis by image skeletonization**” 4<sup>th</sup> IEEE Workshop on application on Computer Vision, 1998, pg. 15-21.
  18. M. Bertozzi, A. Broggi, S. Castelluccio, “**A real-time oriented system for vehicle detection**” Special quintuple issue: Euromicro 1995 short contributions, volume 43, 1997, pg.317 -325.
  19. Y. Du, N.P. Papanikolopoulos, “**Real-time vehicle following through a novel symmetry-based approach**” Proceedings on IEEE International Conference on Robotics and Automation, volumen 4, 1997, pg. 3160-3165.
  20. C. Hoffmann, T. Dang, C. Stiller, “**Vehicle detection fusing 2D visual features**” 2004 IEEE Intelligent Vehicles Symposium, Volumen 14, junio 2004, pg. 280-285.
  21. Hu, Z., and K. Uchimura, “**Tracking cycle: a new concept for simultaneous tracking of multiple moving objects in a typical traffic scene**” EURASIP Journal on Applied Signal Processing, 2005, pg. 2322-2329.
  22. Huang, S.-S., C.-J. Chen, P.-Y. Hsiao, L.-C. Fu, “**On-board vision system for lane recognition and front-vehicle detection to enhance driver's awareness**” 2004 Proceedings of ICRA IEEE International Conference on Robotics and Automation, volumen 3, 2004 pg. 2456-2461.
  23. Wu, J. and X. Zhang, “**A PCA classifier and its application in vehicle detection**” 2001 Proceedings of International Joint Conference on Neural Networks, IJCNN, volumen 1, 2001, pg. 600-604.
  24. S.-W. Seol, J.-H. Jang, H.-S. Kim, C.-H. Lee, and K.-G. Nam, “**An automatic detection and tracking system of moving objects using double difference based motion estimation**” Proceedings of International Technical Conference on Circuits/Systems, Computers and Communications, 2003, pg. 260-263.
  25. J. Suhr, J. Hong, “**Automatic free parking space detection by using motion stereo-based 3D reconstruction**” Machine Vision and Applications, Springer Berlin / Heidelberg, Berlín, Alemania, Julio 2008.
  26. A.Argyros, S. Orphanoudakis, “**Independent 3D Motion Detection Based on depth elimination in normal flow fields**” IEEE Computer Society Conference Proceedings of Computer Vision and Pattern

Revisado OK. 08 / Febrero / 2009

Dr. Boris X. Vintimilla Burgos





# ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA



- Recognition, Revista 17, Jun 1997 pg. 672-677.
27. W. Grimson, C. Stauffer, R. Romano, **“Using adaptive tracking to classify and monitor activities in a site”** IEEE Computer Society Conference Proceedings of Computer Vision and Pattern Recognition, Revista 18, Jun 1998 pg. 22-29.
  28. D. Beymer, P. McLauchlan, B. Coifman, J. Malik **“A real time computer vision system for measuring traffic parameters”** IEEE Computer Society Conference Proceedings of Computer Vision and Pattern Recognition, Revista 17, Jun 1997 pg. 495-501.
  29. T. Svoboda **“Kanade Lucas Tomasi tracker”** Proceedings of 13th International Conference on Computer Communications and Networks ICCCN, Revista 13, Agosto. 2004 pg.904-909
  30. M. Shih, Y. Chang, B Fu, C Huang **“Motion-based Background Modeling for Moving Object Detection on Moving Platforms”** Proceedings of 16th International Conference on Computer Communications and Networks ICCCN, Revista 16, Aug. 2007 pg.1178-1182
  31. A. McIvor **“Background Subtraction Techniques”** Proceedings of Image and Vision computing, Reveal Limited, Auckland Nueva Zelanda, 2000.
  32. C. Stauffer, W. Grimson **“Adaptive background mixture models for real-time tracking”** 1999. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999, Vol. 2.
  33. D. Koller, K. Daniilidis, H.-H. Nagel **“Model-Based Object Tracking in Road Traffic Scenes”** International Journal of Computer Vision 10:3, 1993, pg. 257-281.
  34. J. Boreczky, L. Wilcox **“A hidden Markov model framework for video segmentation using audio and image”** Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, 1998, pg. 3741-3744.
  35. C. Anderson, P. Burt, G. van del Wal **“Change detection and tracking using pyramid transformation techniques”** SPIE – Intelligent robots and computer vision, vol. 579, 1985, pg. 72-78.
  36. J. Barron, D. Fleet, S. Beasuchemin **“Performance of optical flow techniques”** Int. J. Computer Vision, vol. 12, enero 1994, pg.42-77.
  37. I. Haritaoglu, D. Harwood, L.S. Davis **“W<sup>4</sup> Who? Where? When? What? A real time system for detecting and tracking people”** FGR98, 1998, pg. 222-227.
  38. G. Cohen, G. Medioni **“Multi-views tracking within and across uncalibrated camera streams”** UC Berkeley First ACM SIGMM international workshop on Video surveillance, 2003, pg. 21-23
  39. P. KaewTraKulPong, R. Bowden **“An Improved Adaptive Background Mixture Model for Realtime Tracking with Shadow Detection”** Proceedings of 2nd European Workshop on Advanced Video Based Surveillance Systems, AVBS01, 2001, pg. 25-28.

Revisado OK. 08 / Febrero / 2009

Dr. Boris X. Vintimilla Burgos