

Conteo Automático de Vehículos Usando *Frame Differencing* para un Sistema Inteligente de Transporte

C.A. Quiceno¹, J.W. Branch²

¹Departamento de Ciencias de la Computación y de la Decisión, Facultad de Minas, Universidad Nacional de Colombia, Medellín, Colombia. Email: cquiceno@unal.edu.co

²Departamento de Ciencias de la Computación y de la Decisión, Facultad de Minas, Universidad Nacional de Colombia, Medellín, Colombia. Email: jwbranch@unal.edu.co

Palabras Clave: Conteo de Vehículos, Volumen Vehicular, *Frame Differencing*, Seguimiento de Vehículos, Sistemas Inteligentes de Transporte, Visión por Computador.

Resumen

Para recolectar información de las vías de una ciudad, los Sistemas Inteligentes de Transporte juegan un papel fundamental ya que son los encargados de capturar, almacenar y analizar toda la información generada por los diferentes dispositivos (detectores magnéticos, tubos de presión, sensores de microondas, pistolas de radar o infrarrojas y análisis de imágenes). El análisis de esta información inicia con la obtención de los parámetros básicos de una vía tales como volumen vehicular, velocidad y densidad. Este trabajo presenta un método para realizar el conteo automático de vehículos en una vía y realizar el cálculo del volumen de la misma utilizando *Frame Differencing* para determinar los objetos en movimiento.

1 Introducción

La congestión del tráfico vehicular es un problema creciente en las zonas urbanas de todas las ciudades del mundo. Gran parte de la congestión es causada por “incidentes”, los cuales se consideran como eventos generados por accidentes, inundaciones, caídas de árboles, postes, entre otros, los cuales generan que la capacidad de una vía disminuya.

En la actualidad, las Centrales de Administración de Tráfico (TMC, Traffic Management Center por sus siglas en inglés), apoyados por los Sistemas Inteligentes de Transporte (ITS Intelligent Transportation System por sus siglas en inglés), son los centros encargados de administrar y recoger la información de las vías de una ciudad. Su objetivo principal es procesar todos estos datos para monitorear el estado de las vías, además permitiendo que esta información sea enviada a los medios de comunicación y conductores.

El uso de técnicas de visión artificial en ITS [1] permite hacer la detección y conteo de vehículos, mediciones de velocidad, clasificación, densidad y volumen vehicular. Permite hacer mediciones sobre uno o varios carriles, definir zonas de detección, entre otros [2]. Como parámetro básico el volumen vehicular se define como la cantidad de vehículos que circulan por una sección de vía durante un periodo de tiempo determinado. Se expresa en vehículos/hora [3, 4].

Diferentes métodos han sido propuestos para realizar el conteo y seguimiento de los vehículos. En [2] se realiza un análisis comparativo (ventajas, desventajas, datos recolectados, costo) de las diferentes tecnologías utilizadas. En [5] se realiza un análisis crítico del estado del arte de las técnicas basadas en visión por computador en video vigilancia urbana. En el conteo automático de vehículos [6, 7] proponen un método eficiente para diferentes condiciones de tráfico obteniendo errores inferiores al 5%, al realizar una mejor segmentación de los vehículos de la vía. Para disminuir el problema de oclusiones en el análisis de imágenes [8] aumenta la precisión del conteo realizando una segmentación basada en texturas para determinar los contornos de los vehículos y eliminar las sombras. Para determinar parámetros como: nivel de ocupación de una vía, [9] plantea una metodología basada en segmentación de vehículos con actualizaciones periódicas para evitar inconvenientes con el cambio de iluminación de la vía, para la velocidad [10] presenta un método basado en las coordenadas del objeto en movimiento y [11] tiene en cuenta la longitud del bloque del vehículo detectado y tiempo que tarda en cruzar las líneas de acumulación.

El objetivo de este trabajo es presentar un método que permita realizar el conteo automático de vehículos utilizando *Frame Differencing* y calcular el volumen vehicular de una vía de manera eficiente y con bajo costo computacional.

2 Metodología propuesta

Un diagrama conceptual es presentado en [8] el cual describe los componentes de un sistema visual de vigilancia de tráfico. Para este trabajo la figura 1 representa el método propuesto para realizar la detección, seguimiento y conteo vehicular.

2.1 Selección de Área de interés (ROI, Region of Interest)

Para realizar la captura de los videos se utilizaron 2 cámaras digitales diferentes (Sony Cybershot DSC-H2 y Samsung SL-420). Teniendo en cuenta [2, 8] para disminuir la oclusión generada por los vehículos la cámara fue ubicada por encima de la vía y alineada con la línea discontinua separadora de carril e inclinada con respecto al plano horizontal de la vía en un ángulo entre $-90^\circ \pm 65^\circ$ [6]. Para delimitar el área de procesamiento del algoritmo y aumentar su precisión y rendimiento se define un ROI el cual contenga los carriles a analizar y se eliminan las aéreas no correspondientes a la vía,

que generen ruido o no brinden información. La figura 2 representa un ejemplo de ROI en la vía.

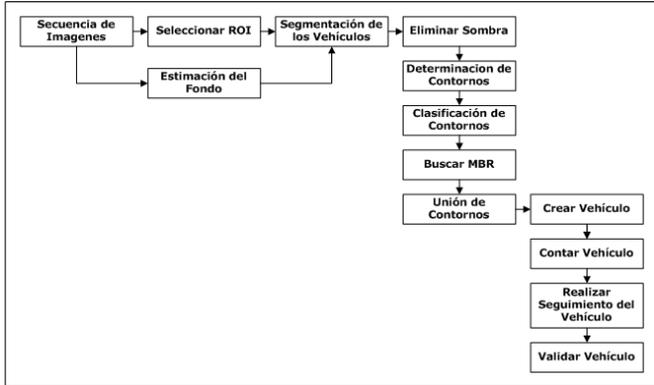


Figura 1. Diagrama de bloques del método propuesto



Figura 2. ROI de la imagen a analizar

2.2 Estimación del Fondo

Trabajos previos [5, 6] realizan una descripción comparativa entre las diferentes técnicas para la obtención del fondo de una escena en el análisis automático de videos de tráfico. A pesar de su alto costo computacional Gaussian Mixture Model (GMM) es una técnica ampliamente utilizada por investigadores en el análisis de tráfico, se encuentra como implementación (CV_BG_MODEL_MOG[12]) dentro de la librería OpenCV [13] y permite obtener el fondo de una escena con objetos en movimiento (Figura 3).

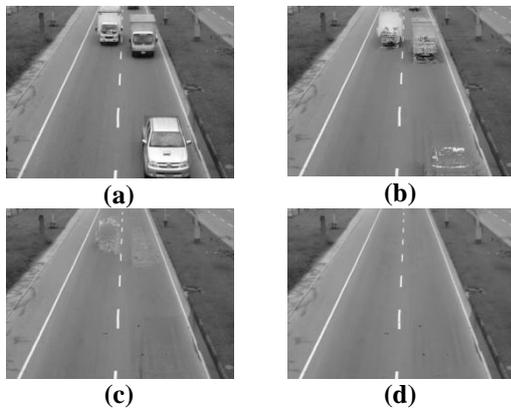


Figura 3. Obtención del fondo paso a paso mediante GMM

2.3 Segmentación de los Vehículos (Binarización)

Frame Differencing es la técnica más rápida para obtener elementos de primer plano [5, 10] de una escena, consiste en

el cálculo pixel a pixel de la diferencia entre 2 imágenes consecutivas mediante la ecuación (1).

$$I_n - I_0 \leq F_n \quad (1)$$

Para aumentar su precisión se usa el fondo obtenido mediante GMM (I_0) como de fondo referencia de la escena y asumiendo que este permanece estático [14], se aplica el operador diferencia con la imagen a analizar (I_n) y para eliminar ruido se umbraliza la imagen obteniendo así los elementos en movimiento de la imagen (F_n). La figura 4 representa la segmentación de los elementos en movimiento de la escena.



Figura 4. Elementos de primer plano de la vía.

2.4 Eliminación de Sombras

En el proceso de segmentación de la imagen el vehículo y su sombra se presentan como un solo elemento porque ambos están en movimiento (Figura 4b). Diferentes métodos han sido descritos y comparados en [15] entre ellos, la detección y eliminación de sombras utilizando el espacio HSV es uno de los más utilizados en sistemas de análisis de tráfico [5]. Se define que un pixel es clasificado como sombra si cumple las 3 condiciones presentadas en la ecuación (2).

$$SP_k(x, y) = \begin{cases} 1 & \text{si } \alpha \leq \frac{I_k^V(x, y)}{B_k^V(x, y)} \leq \beta \\ & \wedge (I_k^S(x, y) - B_k^S(x, y)) \leq \tau_S \\ & \wedge |I_k^H(x, y) - B_k^H(x, y)| \leq \tau_H \\ 0 & \text{Cualquier otro} \end{cases} \quad (2)$$

La figura 5 presenta como se identifican 2 vehículos antes y después de aplicar el proceso de eliminación de sombras.



Figura 5. Proceso de eliminación de sombras.

2.5 Creación de Bloques

Después de realizar el proceso de binarización y eliminación de sombras [6, 11] propone un método para unir o dividir los bloques basados en la distancia entre los centros y el ancho del bloque. Para la creación de los candidatos a vehículos, los bloques obtenidos deben ser clasificados (se filtran por área, relación alto/ancho y ancho mínimo). En caso de existir varios bloques en la misma escena, deben ser agrupados por

su proximidad o inclusión, basados en un conjunto de reglas para obtener el bloque final correspondiente a un vehículo. La figura 6 presenta la clasificación, el conjunto de reglas y los resultados obtenidos para cada una de ellas.

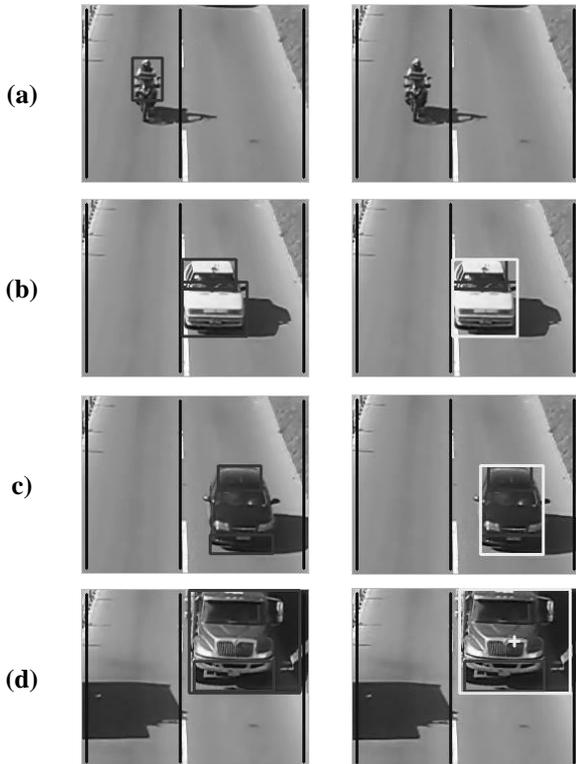


Figura 6. Reglas para la clasificación y unión de bloques. (a) Eliminación por área mínima, ancho mínimo o relación alto/ancho, (b) unión por solapamiento, (c) distancia entre centros y (d) inclusión

2.6 Seguimiento y Conteo de Vehículos

Para realizar el seguimiento de un vehículo se extrajeron las características correspondientes a centros del bloque (coordenadas X_c y Y_c) mediante el cálculo de momentos [16], al bloque se le adiciona la información de largo, ancho y carril. Se crea un bloque tipo vehículo con esta información y se almacenan en un vector para comparar los bloques encontrados en cada una de las imágenes y determinar el vehículo al cual pertenecen o si es un nuevo vehículo. Para clasificar un bloque como nuevo vehículo se debe cumplir las condiciones de la ecuación (3):

$$V_{nuevo} = \{Y_c > Y_{ini} \wedge B_i \notin V_n\} \quad (3)$$

Donde:

- V_n : Vector de vehículos.
- Y_c : Posición Y del centro del bloque.
- Y_{ini} : Posición Y inicial para detección.
- B_i : Bloque de la escena.

En el ROI se definen 3 regiones para realizar el análisis de detección de un nuevo vehículo, seguimiento y finalización del seguimiento. La figura 7 y 8 representan las zonas descritas en el proceso de seguimiento de un vehículo.



Figura 7. Zonas de detección, seguimiento y finalización.

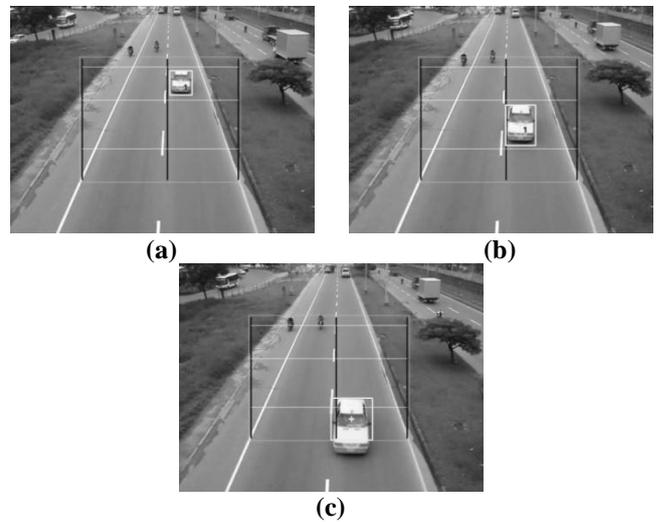


Figura 8. Proceso Seguimiento (a) Detección y conteo, (b) Seguimiento y (c) finalización (Validación del Vehículo)

Para realizar el seguimiento de un vehículo nuevo (V_{nuevo}) el bloque clasificado en las imágenes siguientes debe estar dentro de la zona de seguimiento (Figura 7) y debe ser encontrado en el vector de vehículos (V_n).

Un vehículo es contado al ser detectado, pero si este no puede ser seguido o finalizado es descartado de V_n .

3 Pruebas

El método propuesto para realizar el conteo de vehículos fue implementado mediante el uso de la librería OpenCV [13] corriendo sobre un computador DELL Studio 1555 con sistema operativo Microsoft Windows 7 Home Premium Edition (64-bit), procesador Intel(R) Core(TM)2 Duo CPU T6600 @ 2.20GHz y memoria RAM de 4GB DDR2-SDRAM (PC2-6400 / 800 MHz). Se tomaron videos en 2 puntos de la Avenida Regional de la ciudad de Medellín. La cámara fue ubicada en los puentes peatonales de las estaciones del Metro Acevedo y Caribe del Metro de Medellín (Figura 9).



Figura 9. Puente Peatonal Estación del Metro de Medellín (a) Estación Acevedo y (b) Estación Caribe.

Los datos de los videos tomados se presentan en la tabla 1.

Video Nro.	Estación	Duración	Conteo Manual	Volumen Vehicular
1	Acevedo	00:02:18	31	809
2	Caribe	00:02:00	29	870
3	Caribe	00:02:00	71	2130
4	Caribe	00:05:17	113	1283

Tabla 1. Datos de conteo y volumen de los videos analizados.

4 Resultados

Los resultados obtenidos del conteo automático de vehículos realizado mediante el método propuesto son presentados en la tabla 2 y figura 10.

Video Nro.	1	2	3	4
Conteo Manual*	31	29	71	113
Conteo Automático*	31	28	70	111
Falsos Positivos*	0	0	1	2
No Contados*	0	1	2	4
Conteo Corregido*	31	28	69	109
Volumen Calculado*	809	840	2070	1238
Error	0.00%	3.45%	2.82%	3.54%
Tiempo Procesamiento**	01:23	01:35	01:54	05:00
Área ROI***	41040	88894	180761	82620

Tabla 2. Resultado del conteo mediante el método propuesto (* vehículos, ** mm:ss, *** pixeles²).

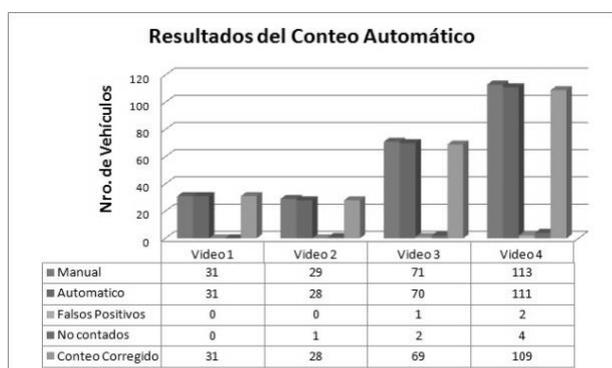


Figura 10. Representación gráfica de los resultados obtenidos.

4.1 Errores Encontrados

4.1.1 Falsos Positivos

Los falsos positivos se producen cuando el método cuenta un vehículo sin estar presente en el ROI o cuando 2 bloques de un mismo vehículo no pueden ser agrupados. Estos errores pueden ser minimizados teniendo en cuenta el seguimiento del bloque detectado (ya implementado en el algoritmo) o realizando ajustes al proceso de creación de bloques. La figura 10 muestra los falsos positivos reportados.



Figura 10. Ejemplos de Falsos Positivos. (a) Vehículo contados 2 veces. (b) Vehículo detectado sin estar en el ROI

4.1.2 Vehículos No Detectados

Este error se presenta cuando el bloque obtenido después del proceso de segmentación de bloques no cumple con los criterios de clasificación para ser identificado como posible bloque de un vehículo. Estos errores pueden ser minimizados haciendo ajustes al proceso de segmentación y eliminación de sombras para aumentar los pixeles identificados como parte del vehículo. La figura 11 muestra un vehículo en la zona de detección y que no fue clasificado como vehículo.



Figura 11. Ejemplos de Vehículos no Detectados

5 Conclusiones

El método propuesto presenta un gran desempeño en el tiempo de procesamiento de las imágenes, ya que es menor que el tiempo de duración del video además se considera eficiente por presentar errores menores al 4% en el conteo de vehículos, lo cual nos permite pensar en que el conteo se pueda hacer en tiempo real haciendo uso de una cámara instalada en un computador portátil o una red de monitoreo de tráfico.

El tiempo de procesamiento es afectado directamente por el tamaño del ROI a analizar, esto se evidencia en los videos 1, 2 y 3 los cuales a pesar de tener una duración aproximada de 2 minutos el tiempo de procesamiento aumenta debido al aumento en el tamaño del ROI

La segmentación de los bloques y eliminación de sombras es uno de los procesos más importantes en el método y debe ser parametrizado con precisión teniendo en cuenta las condiciones de iluminación para así evitar los inconvenientes reportados como errores.

6 Trabajo a futuro

A partir de los datos obtenidos de los centros de los bloques clasificados como vehículos y el seguimiento se puede obtener la velocidad de cada uno de ellos y así calcular los demás parámetros de la vía (Flujo, densidad y Velocidad).

Como trabajo adicional se puede considerar la opción de realizar pruebas en escenarios con baja iluminación (noche, lluvia, etc), en momentos de congestión y determinar la eficiencia del algoritmo.

Referencias

- [1] K. S. Jie y M. Liu, "Computer vision based real-time information acquisition for transport traffic," *Proceedings of the 2005 IEEE International Conference on Information Acquisition*, pp. 164-169, 2005.
- [2] US Department of Transportation, Traffic detector handbook, vol. 1, Federal Highway Administration, 2006, p. 290.
- [3] J. J. Posada y A. B. Mejía, "La velocidad de recorrido en vías urbanas de acuerdo con el volumen y la jerarquía vial - Caso Envigado -," *Revista DYNA*, n° 130, pp. 31-37, 2000.
- [4] N. J. Garber y L. A. Hoel, *Ingeniería de tránsito y de carreteras*, 3a ed., Thomson, 2005.
- [5] N. Buch, S. A. Velasti y J. Orwell, "A review of computer vision techniques for the analysis of urban traffic," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, n° 3, pp. 920-939, 2011.
- [6] F. C. Calderon y G. E. Urrego, *Conteo automático de vehículos*, Bogota, 2008, p. 62.
- [7] L. Jingwei, Z. Yong, Y. Yule, L. Wei y L. Kai, "Vehicle capturing and counting using a new edge extraction approach," de *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, 2011.
- [8] C. C. Pang, W. W. Lam y N. H. Yung, "A Method for Vehicle Count in the Presence of Multiple-Vehicle Occlusions in Traffic Images," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, n° 3, pp. 441-459, 2007.
- [9] E. Ospina, E. Tasco, J. Valencia y C. Madrigal, "Traffic flow control using artificial vision techniques," de *Computing Congress (CCC), 2011 6th Colombian*, 2011.
- [10] H. A. Rahim, U. U. Sheikh, R. B. Ahmad, A. S. Zain y W. N. Ariffin, "Vehicle speed detection using frame differencing for smart surveillance system," de *Information Sciences Signal Processing and their Applications (ISSPA), 2010 10th International Conference on*, 2010.
- [11] F. C. Calderón, *Sistema de adquisición de parámetros de tráfico vehicular*, Bogota DC, 2010, p. 58.
- [12] OpenCVWiki, "The OpenCV Video Surveillance / Blob Tracker Facility," 2009. [En línea]. Available: <http://opencv.willowgarage.com/wiki/VideoSurveillance> [Último acceso: 2012].
- [13] OpenCV, "Open Source Computer Vision Library OpenCV," [En línea]. Available: <http://sourceforge.net/projects/opencvlibrary>. [Último acceso: 2012].
- [14] W. L. Lam, C. C. Pang y N. H. Yung, "Highly accurate texture-based vehicle segmentation method," *Optical Engineering*, vol. 43, n° 3, pp. 591-603, 2004,.
- [15] A. Prati, I. Mikic, M. M. Trivedi y R. Cucchiara, "Detecting moving shadows: algorithms and evaluation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, n° 7, pp. 918- 923, 2003.
- [16] L. Kotoulas y I. Andreadis, "Image analysis using moments," de *5th Int. Conf. on Technology and Automation*, Thessaloniki, Greece, 2005.