

## Inspección Visual Automática mediante Segmentación de Color y una cámara de bajo costo

Rafael VALENCIA CARREÑO

Alfredo SANTANA DÍAZ

Escuela de Ingeniería y Arquitectura

Instituto Tecnológico y de Estudios Superiores de Monterrey, campus Toluca.

Eduardo Monroy Cárdenas #2000, San Antonio Buenavista. Toluca, Edo. De México CP. 50110

Tel: (722) 2799990

{ a00788663, asantana }@itesm.mx

**Resumen** – En este artículo se describe una metodología de procesamiento de imágenes, para llevar a cabo la inspección visual automática de productos, orientada hacia aplicaciones de inspección en las que sea conveniente el uso de discriminación de color sin restricciones de precisión. La metodología se emplea en una aplicación de software que realiza la inspección del envasado de píldoras en un blister<sup>1</sup> –determina el número de píldoras de un color específico y estima la redondez. Las etapas de la metodología comprenden a la adquisición de la imagen –a través de una cámara Web–, el filtrado espacial, la segmentación de color mediante el espacio de color HSI, el filtrado morfológico, el etiquetado de regiones y la extracción de características. Los resultados de las pruebas realizadas bajo condiciones de operación controladas, como iluminación, siempre fueron correctos.

### I. INTRODUCCIÓN

La inspección manual presenta diversos problemas que pueden afectar a la calidad del producto [5]. Una alternativa a la inspección manual es la inspección visual automática. Ésta permite realizar una inspección rutinaria, limpia, segura, versátil y rápida de ciertas características de productos; adicionalmente, hace posible inspeccionar en condiciones adversas y durante periodos largos [8].

En el mercado se encuentran diversos productos para la inspección visual automática. Sin embargo, estos productos pueden representar un gasto elevado para algunas empresas. Además, el costo de la aplicación aumenta si se necesita examinar el color de objetos industriales.

No obstante lo anterior, el costo no siempre debe de ser alto en una aplicación de inspección de color [1]. Primero, existen muchas aplicaciones en donde no se requiere discriminar colores con precisión (v. gr., en el empaqueo de crayones de color, para verificar que cada paquete incluya los colores correctos). Segundo, en tales

aplicaciones, existe una área de oportunidad para reducir gastos al emplear cámaras a color de bajo costo: cámaras que utilicen sensores ópticos CMOS.

En el presente trabajo se propone una metodología de procesamiento de imágenes para llevar a cabo la inspección visual automática de productos, en aplicaciones donde no se requiere discriminar colores con precisión, con el propósito de reducir el costo monetario. La metodología propuesta es empleada en una aplicación de software, con el objetivo de realizar la inspección del envasado de píldoras en un blister.

Con base en una imagen del blister, la aplicación determina si el recipiente contiene el número correcto de píldoras de un color específico –se eligió, para este caso, de color verde– y estima la redondez de cada píldora en el blister –obtiene área y perímetro. El blister puede estar en cualquier posición dentro del cuadro, lo que hace a la aplicación invariante ante rotaciones o traslaciones del producto.

La adquisición se hace por medio de una cámara de bajo costo –cámara Web– y los algoritmos de procesamiento de imágenes fueron escritos en lenguaje C e implementados en Visual C++ 6.0.

Se establecieron las siguientes condiciones de operación: iluminación blanca, cámara fija, distancia entre la cámara y el objeto de inspección también fija y fondo sin objetos que tengan el mismo color que el de las píldoras de interés.

### II. METODOLOGÍA

En la fig. 1 se muestra un diagrama de bloques que resume el desarrollo metodológico. Cada uno de los procesos indicados en la fig. 1 se detallarán a lo largo de la sección.

Inicialmente, una imagen –320x240, con profundidad de 24 bits– es adquirida de la escena. La imagen adquirida se muestra en pantalla y se pasa a las funciones de filtrado. La imagen filtrada es segmentada para extraer a los objetos de interés; sin embargo, dado que esta imagen puede contener ruido, es necesario

<sup>1</sup> Envase para manufacturados pequeños que consiste en un soporte de cartón o cartulina sobre el que va pegada una lámina de plástico transparente con cavidades en las que se alojan distintos artículos.

filtrarla. El filtrado de la imagen segmentada se hace por medio de operaciones morfológicas.

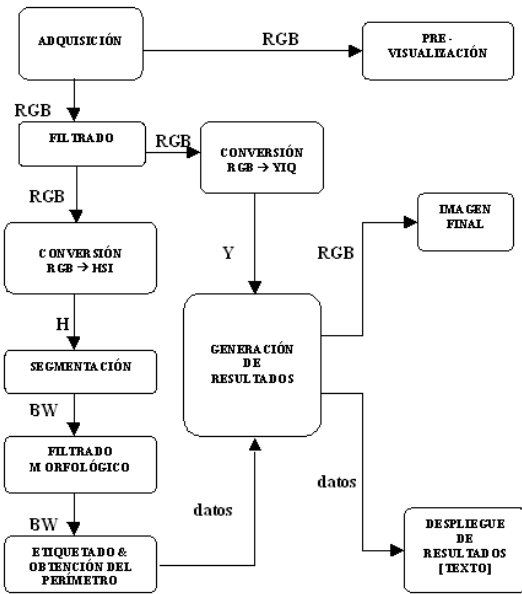


Fig. 1 Diagrama de bloques

Después del filtrado morfológico, la imagen segmentada únicamente tendrá a los objetos de interés, los cuales son etiquetados y se obtiene el perímetro de cada uno. Esta información es necesaria para la generación de resultados. Adicionalmente, para generar los resultados, se obtiene una imagen de brillo de la imagen original.

Los resultados consisten en una versión en escala de grises (componente Y del espacio de color YIQ) de la imagen adquirida, con los objetos detectados enmarcados con un cuadro de color rojo, y en el despliegue del texto, que indica el número de objetos detectados y la redondez de cada una de las píldoras.

#### A. Filtrado espacial

En este proceso fueron empleados filtros de suavizado para eliminar ruido. Como este tipo de filtros hace borrosa a la imagen para eliminar el ruido[7], se buscó un tipo de filtro que conservara mejor la forma de los objetos.

Se probaron tres alternativas: tres repeticiones de filtros pasa bajos lineales, cuya máscara (de 3x3) proporciona mayor peso al píxel central; tres iteraciones de filtros de mediana, con máscara de 3x3; y una combinación de los anteriores: dos filtros de mediana seguidos de un filtro pasa bajos. La opción elegida fue la que emplea solamente filtros de mediana.

Cada una de las opciones fue evaluada para determinar qué proceso de filtrado conserva mejor la forma de los objetos y cuál es más efectiva en la eliminación del ruido.

Para conocer qué filtro conserva mejor la forma de los objetos se empleó un análisis de histograma de las imágenes filtradas. Se utilizó una imagen sintética de un patrón de tablero de ajedrez. Esta imagen tiene un

histograma sencillo, únicamente presenta dos funciones delta en los niveles de gris de 100 –cuadrados oscuros– y 200 –cuadrados claros–. La imagen original fue filtrada mediante los tres procedimientos. En la fig. 2 se muestran los histogramas de las imágenes filtradas.

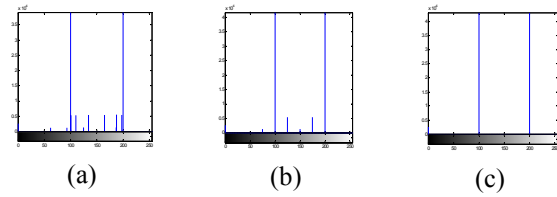


Fig. 2 Histogramas de las imágenes filtradas con: filtros pasa bajos(a), filtros de mediana y pasa bajos (b) y filtros de mediana(c)

El histograma del filtrado pasa bajos incluye a las dos funciones delta esperadas y picos entre las dos funciones, tales picos se deben a que este filtrado hace borrosos a los bordes de la imagen. El procedimiento que emplea filtros de mediana y pasa bajos presenta también picos entre las funciones delta; aunque presenta menos picos que en el filtrado anterior, este filtrado también provoca un suavizado en los bordes. En el histograma de filtros de mediana únicamente se observan las dos funciones delta, lo cual indica que preserva mejor la forma de los objetos de la imagen original.

Por otro lado, para conocer qué tan bien es removido el ruido, una imagen fue corrompida mediante dos tipos de ruido –similares a los que presenta la cámara–, posteriormente, la imagen ruidosa fue filtrada con los tres procedimientos. Se calculó el error medio cuadrático entre la imagen filtrada y la imagen original.

Los ruidos añadidos fueron: ruido blanco Gaussiano de media cero con varianza de 0.1 y ruido sal y pimienta.

La Ecuación (1) expresa al error medio cuadrático, donde  $y_i$  es el  $i$ -ésimo nivel de gris del píxel de la imagen filtrada y  $x_i$  es el nivel de gris del  $i$ -ésimo píxel de la imagen original.

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - x_i)^2, \quad (1)$$

En la tabla 1 se presentan los resultados obtenidos en el cálculo del error medio cuadrático. Los valores están normalizados con respecto al error entre la imagen ruidosa y la imagen original.

En la tabla 1 se observa que el filtrado que mejor elimina el ruido Gaussiano fue el que emplea filtros pasa bajos lineales. Sin embargo, no hay gran diferencia entre los errores; el error del filtrado lineal es 1.45% menor que el del filtrado que utiliza filtros de mediana.

Para el caso del ruido sal y pimienta, el procedimiento que emplea únicamente los filtros de mediana fue el que presenta el mejor desempeño; su error es 10.03% menor al que presenta el filtrado lineal.

TABLA I  
VALORES DEL ERROR MEDIO CUADRÁTICO

	Ruido Gaussiano	Ruido sal y pimienta
Imagen con ruido	100 %	100 %
Imagen filtrada con los filtros pasa bajos	14.35 %	19.15 %
Imagen filtrada con los filtros de mediana y pasa bajos	15.33 %	10.36 %
Imagen filtrada con los filtros de mediana	15.80 %	9.12 %

Por último, mediante las pruebas presentadas se concluye que, de las tres opciones mostradas, el filtrado mediante filtros de mediana presenta mejor conservación de la forma de los objetos y remueve el ruido de una manera más efectiva.

### B. Segmentación

La segmentación consiste en separar a los objetos de interés del resto de los objetos de la imagen. En este caso, los objetos de interés consisten en píldoras verdes, cuya característica que los distinguen del resto de los objetos es su color –se asume que no hay más objetos verdes en la escena. De esta manera, cuando se separe a todos los objetos verdes del cuadro, se estará segmentado a todas las píldoras de interés.

El color de los objetos fue especificado por medio del matiz. Por lo tanto, el proceso de segmentación se llevo a cabo en el espacio HSI, considerando únicamente a la componente de matiz (H).

El primer paso para segmentar a las imágenes consistió en la obtención del matiz a partir de las componentes RGB de la imagen. En (2) se expresa esta operación [7].

$$H = \cos^{-1} \frac{0.5[(R-G)+(R-B)]}{\sqrt{(R-G)(R-G)+(R-B)(G-B)}}, \quad (2)$$

Sin embargo, es posible simplificar la obtención del matiz. En (3) se muestra la fórmula de Bajon [14] para obtener el matiz, a partir de las componentes RGB. En esta ecuación ya no son empleadas funciones trigonométricas.

$$H = \begin{cases} \text{Acromático,} & \text{Si } R = G = B \\ H = \frac{G-B}{3(R+G-2B)}, & \text{Si } B = \min(R, G, B) \\ H = \frac{B-R}{3(G+B-2R)} + \frac{1}{3}, & \text{Si } R = \min(R, G, B) \\ H = \frac{R-G}{3(R+B-2G)} + \frac{2}{3}, & \text{Si } G = \min(R, G, B), \end{cases} \quad (3)$$

Por su sencillez, (3) fue implementada para obtener el matiz. El resultado de (3) consiste en una imagen de matiz  $H(x, y)$  cuyos píxeles están en intervalo de 0 a 1.

Así, el matiz correspondiente al color verde está, aproximadamente, entre 0.3 y 0.5. Una vez que se calculó el matiz, se obtuvo una imagen binaria  $BW(x, y)$  mediante (4).

$$BW(x, y) = \begin{cases} 255, & \text{si } 0.3 \leq H(x, y) \leq 0.5 \\ 0, & \text{en otro caso} \end{cases}, \quad (4)$$

En la fig. 3 se muestran los resultados del proceso de segmentación. Se observa que en la imagen binaria existen píxeles blancos que no pertenecen a los objetos de interés; es decir, la imagen binaria presenta ruido. Sin embargo, este ruido puede ser eliminado mediante el filtrado morfológico.

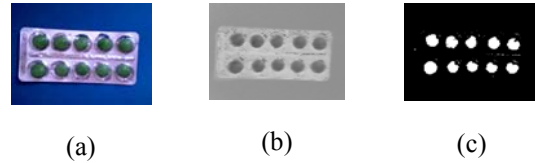


Fig. 3 (a) Imagen adquirida, (b) matiz y (c) segmentación de las píldoras.

### C. Filtrado morfológico

En este proceso se limpia a la imagen binaria, obtenida en la segmentación, por medio de operaciones morfológicas binarias. Las operaciones morfológicas binarias buscan mejorar o clarificar la estructura de los objetos que han sido discriminados en una imagen binaria [3, 7].

Fueron probados tres métodos de filtrado: tres erosiones con un elemento de estructura de 3x3, apertura con un elemento de estructura de 3x3 y, por último, tres erosiones con un elemento de estructura de 3x3, seguido de tres dilaciones con un elemento de estructura de 5x5 –este último procedimiento es el que fue empleado en este trabajo.

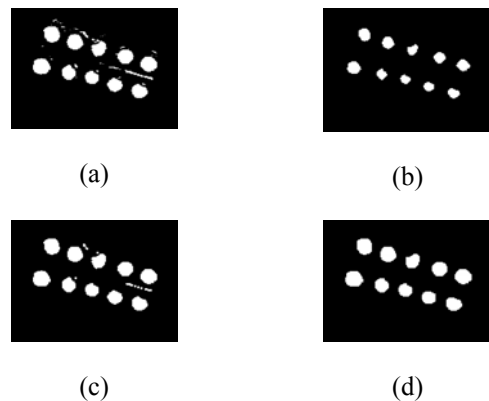


Fig. 3 (a) Imagen adquirida, (b) matiz y (c) segmentación de las píldoras.



Fig. 4 (a) Imagen con ruido. (b) Resultado de erosionar (a) con (e) -3 iteraciones. (c) Apertura de (a) con (e). (d) Resultado de dilatar (b) con (f) -3 iteraciones.

#### D. Etiquetado

Para contar a los objetos segmentados, se implementó un algoritmo de etiquetado de componentes conexos.

El objetivo principal del algoritmo de conexión es determinar si los píxeles forman parte de un objeto. Este algoritmo trabaja sobre la vecindad de un píxel para determinar cuáles de sus vecinos están conectados con él [9, 12].

El algoritmo implementado asume que hay grandes cantidades de memoria y puede etiquetar a todos los objetos de la imagen en dos barridos. Este algoritmo ayuda en situaciones en las que el número de regiones es pequeño o cuando se conoce previamente el número de regiones que se pueden presentar; así, se asignó sólo la memoria necesaria para la aplicación. Dillencourt [4] presenta una descripción detallada de cómo implementar el algoritmo de etiquetado de forma óptima.

El algoritmo etiqueta a las regiones bajo la premisa de que existe conexión con 4 vecinos y emplea una tabla de equivalencias. En (5) se muestra un ejemplo de la tabla de equivalencias empleada. Los unos indican que existe una equivalencia (v. gr., las etiquetas  $a$  y  $b$  son equivalentes) y los ceros indican la ausencia de la misma.

$$B = \begin{matrix} & a & b & c & d & e \\ \begin{matrix} a \\ b \\ c \\ d \\ e \end{matrix} & \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix} \quad (5)$$

Para etiquetar a las regiones se recorre a la imagen por renglones -de arriba hacia abajo y de izquierda a derecha.

Sean los píxeles  $q$  y  $z$  los vecinos norte y este del píxel  $p$ , respectivamente. Se asigna un número o etiqueta al píxel  $p$  de la siguiente manera:

- Si  $q$  y  $z$  no tienen etiqueta, el píxel  $p$  tendrá una nueva etiqueta.
- Si solamente  $q$  ó  $z$  tiene etiqueta, el píxel  $p$  tendrá la misma etiqueta que la de su vecino.
- Si tanto  $q$  como  $z$  tienen etiquetas, se dice que existe una colisión entre etiquetas; en este caso, el píxel  $p$  adquiere la etiqueta menor y se marca la equivalencia en la tabla.

Después del primer barrido, en la tabla de equivalencias -visualizando a los conjuntos de equivalencias como una relación matemática- los conjuntos de equivalencias se convierten en *relaciones de equivalencia*; es decir, en relaciones que cumplan con las propiedades de *reflexión*, *simetría* y *transitividad*. De esta forma, es posible agrupar a las relaciones de equivalencia en  $k$  conjuntos disjuntos o *clases de equivalencia* [10].

Así, en el momento de registrar una equivalencia en la tabla, también se le aplica la propiedad de simetría. Con respecto a la propiedad de *transitividad*, es posible generar los elementos implícitos mediante la operación de *cerradura transitiva*. Para llevar a cabo la cerradura transitiva se utilizó el algoritmo desarrollado por Stephan Warshall [18].

Finalmente, se hace el segundo barrido, en el sentido inverso al barrido original, mientras se recorre la tabla de etiquetas de la imagen y se reemplazan las etiquetas con base en las clases de equivalencias.

#### E. Redondez

Adicionalmente al número de píldoras presentes en el *blíster*, otra característica que fue extraída de la imagen fue una estimación de la redondez de las pastillas. Para obtener la redondez, de acuerdo con (6), es necesario conocer el área y el perímetro de los objetos.

$$redondez = 4\pi \frac{\text{área}}{\text{perímetro}^2} \quad (6)$$

El área se estima con los marcos de cada píldora; mediante el producto de la base por la altura. El perímetro se obtiene a través de un algoritmo de trazado de contornos. El algoritmo empleado fue el de trazado de vecindad de Moore [17].

Este algoritmo no visita el interior del perímetro, en el caso de que el objeto de interés tenga huecos. Podría haber casos en los que se pierdan algunos píxeles del perímetro, sobre todo cuando la vecindad sea de conexión a cuatro. Sin embargo, comúnmente, el ancho de la vecindad es mucho mayor a un píxel, por lo que el algoritmo tiene mayores probabilidades de éxito.

### III. RESULTADOS

Bajo condiciones normales de operación el programa siempre detectó el número correcto de píldoras y estimó adecuadamente su redondez.

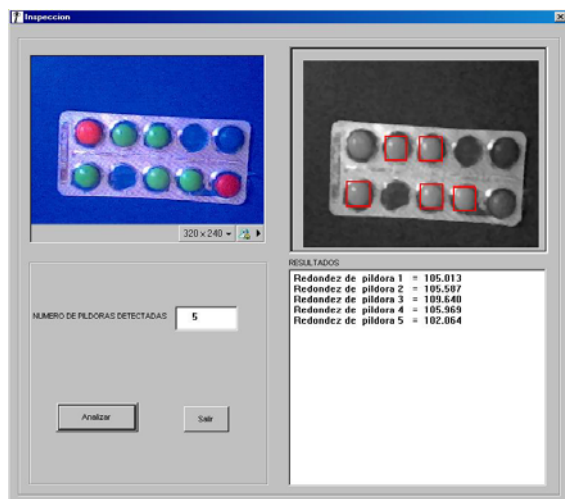


Fig. 5 Aplicación desarrollada en Visual C++

En la fig. 5 se presenta la ventana de la aplicación implementada en Visual C++. En ella se muestra la imagen adquirida, a la izquierda, y la imagen procesada, a la derecha, con las píldoras detectadas enmarcadas en cuadros rojos. También se incluye un mensaje de texto en el que se resumen los resultados del procesamiento de la imagen: se indica el número de píldoras detectada en el *blister* y se despliega la redondez de cada una de ellas.

#### IV. CONCLUSIONES

La inspección manual presenta diversos problemas que afectan a la calidad del producto. Una alternativa a la inspección manual es la inspección visual automática. Actualmente, diversas empresas se dedican a la elaboración de este tipo de sistemas. Sin embargo, algunos de estos productos resultan costosos, sobre todo en aplicaciones de inspección de color.

En el presente trabajo se propuso una metodología para realizar la inspección visual automática de productos que reduce el costo monetario mediante el uso de una cámara de bajo costo. La metodología propuesta se utilizó en una aplicación de *software* implementada en Visual C++, con el objetivo de realizar la inspección del envasado de píldoras en un *blister*. Bajo condiciones normales de operación, el programa siempre detectó el número correcto de píldoras y estimó adecuadamente su redondez.

Existen diversas formas de extender el presente trabajo. La metodología propuesta se puede implementar en un sistema inmerso o en algún otro dispositivo –como un PDA. Otra opción similar, consiste en el uso de *hardware* re-configurable. El trabajo de Bonato *et al* [2] es un buen ejemplo de una implementación de procesamiento de imágenes en *hardware* re-configurable.

La metodología también puede ser extendida con otras técnicas de filtrado. Dado que la imagen adquirida con la cámara de bajo costo presenta ciertos problemas, es conveniente explorar otras técnicas de filtrado digital

[15], como puede ser el uso *wavelets* [16] para el filtrado espacial, así como el uso de filtros adaptivos. Asimismo, es posible incluir en la metodología cierta robustez ante cambios en el color e intensidad de la fuente de iluminación. Lo anterior podría lograrse por medio de una calibración dinámica de los colores, otros espacios de color (v. gr., CIELAB) o variaciones en los modelos de color [6, 11, 13].

#### V. REFERENCIAS

- [1] Batchelor, B., and P.F Whelan, *Intelligent Vision Systems for Industry*, Springer-Verlag London Ltd., UK. 2002.
- [2] V. Bonato, A.K. Sanches, M.M. Fernández, M. P. Cardoso, E. D. V. Simoes and E. Marques, “A Real Time Gesture Recognition System for Mobile Robots”, *ICINCO(2)*, 2004.
- [3] Díaz de leon, J. L., C. Yáñez márquez, *Introducción a la morfología matemática de conjuntos*, IPN(CIC)-UNAM-FCE, México, 2003.
- [4] M.B.Dillencourt, H. Samet, M. Tamminen, “A General Approach to Connected-Component Labeling for Arbitrary Image Representations”, *Journal of the ACM*, Vol. 39, No. 2, April 1992.
- [5] Evans, J.R. and W.M. Lindsay, *The Management and Control of Quality*, 4<sup>th</sup> Edition, South Western College Publishing, USA, 1999.
- [6] G. Finlayson and G. Schaefer, “Hue that is invariant to brightness and gamma”, School of Information Systems, University of East Anglia, UK, 2000.
- [7] Gonzales, R.C. and R. E Woods, *Digital Image Processing*, 2<sup>nd</sup> Edition, Prentice Hall, USA, 2002.
- [8] Graves, M. and B. Batchelor (Eds.), *Machine Vision for the Inspection of Natural Products*, Springer-Verlag London Ltd., UK, 2003.
- [9] Haralick, R.M and L.G. Shapiro, *Computer and Robot Vision, Vols. I and II*, Addison-Wesley, Reading, MA., USA, 1993.
- [10] Hoffman, K. and R. Kunze, *Linear Algebra*, 2<sup>nd</sup> edition, Prentice-Hall, USA, 1971.
- [11] Y. B. Lee, B. J. You, and S. W. Lee, “A real-time Color-based Object Tracking robust to Irregular Illumination Variations”, *Proceedings of the 2001 IEEE International Conference on Robotics & Automation*, Seoul, Korea 2001.

- [12] Pajares, G., J. M. De La Cruz, *Visión por computador: imágenes digitales y aplicaciones*, RA-MA. España, 2001.
- [13] J.B. Park, Y. Yoon, and A.C Kak, "A New Color Representation for Object Tracking under Non-white Illumination Conditions", *Robot Vision Laboratory*, Pardue University, USA, 2003.
- [14] Sangwine, S.G. and R.E.N. Horne (Eds.), *The Colour Image Processing Handbook*, Chapman & Hall, UK, 1998.
- [15] A. Santana-Diaz; N. Hernandez-Gress; D. Esteve, "Digital filtering: application on the driver's impairment detection", *Intelligent Transportation Systems Proceedings*. 2000 IEEE 1-3 Oct. 2000.
- [16] A. Santana-Diaz; B. Jammes; D. Esteve; M.Gonzalez-Mendoza, "Driver hypovigilance diagnosis using wavelets and statistical analysis", *The IEEE 5th International Conference on Intelligent Transportation Systems*, 2002.
- [17] Toussaint, G., *Pattern Recognition Course Notes: Grids, connectivity and contour tracing*, McGill University Montreal, Quebec, 1997.
- [18] S. Warshall, "A theorem on Boolean matrices", *Journal of the ACM*, 9(1):11-12, 1962.