

Segmentación de objetos en movimiento por flujo óptico y color sin información a priori de la escena

Angel Juan Sánchez García y Homero Vladimir Ríos Figueroa

Facultad de Física e Inteligencia Artificial
Universidad Veracruzana, Xalapa, Veracruz, México.
{angesanchez, hrios}@uv.mx

Resumen Una de las tareas más importantes en el procesamiento de imágenes es la segmentación de objetos. En este artículo una metodología propuesta para la segmentación de objetos en movimiento a partir del flujo óptico y el color, sin requerir información a priori de la escena, es presentada. La estimación del flujo óptico es realizada mediante el algoritmo de Lucas y Kanade de manera piramidal. Los vectores del flujo óptico son agrupados por cercanía, dirección y magnitud. La segmentación del objeto es realizada a partir de la cubierta convexa de los puntos confiables del flujo óptico.

Palabras clave: flujo óptico, segmentación, puntos de interés, color, cubierta convexa.

1. Introducción

En la actualidad, existen aplicaciones cuyo funcionamiento dependen de que un observador, auxiliado de una cámara, sea capaz de identificar los objetos que están en su ambiente. Estas aplicaciones se encuentran en áreas como la robótica, la medicina asistida, el control de tráfico vehicular entre otras. La identificación de objetos en una imagen, suele hacerse por 3 características principales: color, forma y textura. Existen casos donde es requerido conocer la ubicación de los objetos de un color en específico, por lo que los pixeles vecinos que comparten un color similar al especificado son agrupados [5, 6, 14, 25]. También los objetos pueden ser identificados en una imagen por una forma predefinida, por ejemplo, encontrar los pixeles que formen un círculo utilizando la transformación de Hough [2, 13]. Por último, regiones suelen ser identificadas si comparten una textura igual o similar, identificada por diversos métodos como en [4, 15]. Sin embargo, si es requerido identificar objetos que se estén moviendo, es necesario tener como mínimo dos imágenes. Lo anterior es debido a que se debe saber si hubo algún cambio en la intensidad de los pixeles entre las dos imágenes, que nos permita identificar un movimiento. Por lo que, aunque existan otros objetos en un video (o secuencia de imágenes), si estos están estáticos, no deberían ser relevantes. Para identificar los objetos que se movieron de una escena a otra, se suele recurrir al flujo óptico. Horn y Schunck en [10] definieron al flujo óptico como la relación de los cambios espacio-

temporal en la intensidad de los pixeles entre dos imágenes, es decir, encontrar la ubicación de los pixeles que cambiaron de intensidad de una imagen a otra. Para el cálculo del flujo óptico, los métodos generalmente pueden ser clasificados en métodos densos como el de Horn y Shunk [10] y poco densos como el de Lucas y Kanade [16]. El método de Lucas y Kanade se basa solo en información local del pixel, derivada de una pequeña ventana del pixel de interés a diferencia del método de Horn y Shunk que se basa en la información global de la imagen. En ocasiones puede ser requerido no solo la identificación de objetos en movimiento, sino también la creación de un modelo que describa el movimiento, es decir, que la computadora reconozca propiedades del movimiento como su velocidad y dirección.

2. Antecedentes

Como primer antecedente, es mencionada la idea de un sistema que pueda “interpretar lo que ve”. Nagel en [19] propone un CVS (por sus siglas en inglés: Cognitive Vision System), con el cual se espera una descripción en lenguaje natural del desarrollo de una escena del mundo real para tener “una comprensión de lo que está pasando”. Sin embargo debe tenerse conocimiento previo en general, pero también conocimientos específicos de cada dominio del discurso, como posiciones, formas o características de las superficies.

Un problema importante en la segmentación por movimiento, ha sido la identificación de discontinuidades en el movimiento. Este problema ha sido atacado desde diferentes enfoques, como el mostrado por Wang and Adelson en [23], uno de los primeros trabajos donde fue propuesto un enfoque iterativo de crear y poner segmentos para obtener mejores resultados. Posteriormente han habido propuestas para maximizar el enfoque anterior [1, 11, 24]. Desafortunadamente la segmentación suele ser impredecible con estos métodos [26].

Una técnica muy usada es combinar información a priori. Mora y colaboradores proponen en [17] la agrupación de puntos de interés mediante su distribución de manera que satisfagan condiciones geométricas definidas a priori.

La segmentación del movimiento auxiliada por color ha dado buenos resultados [3, 9, 12, 18]. De hecho esta combinación ha sido tan exitosa que es utilizada ampliamente en la segmentación de objetos en visión estéreo [22]. Por lo que revisando los trabajos propuestos para segmentar objetos en movimiento, es observado que el color puede ser una herramienta útil para este objetivo.

3. Propuesta

La metodología propuesta es presentada aplicándola a cada par de imágenes en un tiempo t (imagen A) y un tiempo $t+1$ (imagen B). Una vez calculado el flujo óptico por el método de Lucas y Kanade [16] entre las imágenes A y B, los vectores del flujo óptico más representativos de los puntos de interés [21] son obtenidos. Para identificar qué vectores pertenecen a una misma región, partimos de la suposición de que dos vectores pertenecen a una misma región si y solo si cumplen tres condiciones.

La primera condición es que se encuentren cercanos en el espacio 2D, la segunda es que tengan dirección similar, y la tercera que tengan magnitud semejante.

Una vez obtenidas las regiones por flujo óptico (agrupando vectores similares) compuestas por los puntos confiables del flujo óptico, es posible delimitar cada región obteniendo la cubierta convexa de la nube de sus puntos. Puesto que el polígono resultante está definido por sus puntos confiables y agrupa dentro de él algunos puntos del objeto, existen otros puntos, que pertenecen al objeto pero que no están dentro del polígono. Para agregar esos puntos a la región y dar forma similar al objeto, es asumido que esos puntos faltantes tienen color similar a los que están dentro del polígono, por lo que una operación morfológica por color es aplicada partiendo de los puntos interiores del polígono.

4. Metodología

Los pasos a seguir para realizar la segmentación antes mencionada son mostrados en la Figura 1 y descritos a continuación.

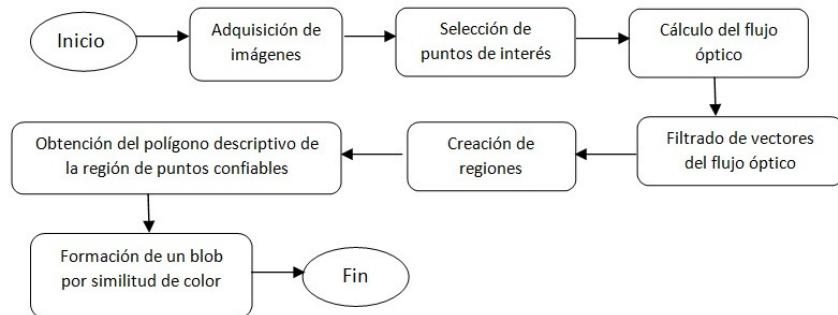


Fig. 1. Diagrama de flujo del algoritmo propuesto.

Para la adquisición de imágenes de los ejemplos mostrados a continuación, fueron adquiridas 10 imágenes por segundo. La resolución de las imágenes adquiridas fue de 2592 x 1944 pixeles.

4.1. Selección de puntos de interés

El cálculo del flujo óptico no puede ser obtenido en todos los pixeles de la imagen debido a que existen pixeles que no nos proporcionan información para ubicarlos en la siguiente imagen (imagen B). Lo anterior es debido a que puede haber pixeles en una superficie de un solo color y sin textura que al ser buscados en la siguiente imagen, varias correspondencias serían obtenidas. Para evitar ese problema, es necesario encontrar pixeles que sean muy diferentes a sus vecinos. A estos pixeles distintivos, Harris [7] los llamó esquinas, cuya característica es tener cambios

considerables de intensidad en direcciones ortogonales. Esta definición está basada en la matriz de derivadas de segundo orden ($\partial^2 x$, $\partial^2 y$, ∂x , ∂y) de las intensidades de la imagen, con lo cual es posible generar la conocida matriz Hessiana para cada pixel. Dado que Harris ocupa el método de mínimos cuadrados, el cálculo suele ser costoso. Shi y Tomassi [21] definen a los puntos como “buenos puntos a seguir” basándose en la definición de la matriz de Harris y tomando como criterio para decidir si lo son, el cálculo de los eigenvalores de esa matriz. Al calcular los eigenvalores λ_1 y λ_2 , uno de tres casos posibles es obtenido.

1. Si $\lambda_1 \approx 0$ y $\lambda_2 \approx 0$, entonces el pixel no tiene información relevante.
2. Si $\lambda_1 \approx 0$ y λ_2 tiene un valor positivo grande, entonces se trata de un borde.
3. Si λ_1 y λ_2 tienen valores positivos grandes, entonces en el pixel hay una esquina.

Si el mínimo eigenvalor supera un umbral α , entonces la esquina tiene un valor confiable para ser identificado. En la Figura 2 es mostrada la imagen A y en círculos rojos los puntos de interés encontrados dicha imagen. Puesto que en la mayor parte de la imagen presentada existe una textura casi de sal y pimienta debido al paisaje presentado, pueden ser obtenidos una gran cantidad de puntos de interés.



Fig. 2. Puntos de interés en color rojo con $\alpha = 0.01$.

4.2. Cálculo del flujo óptico

Una vez que los puntos confiables son obtenidos, es necesario calcular el flujo óptico, es decir, encontrar la correspondencia de esos puntos en la siguiente imagen. Para el cálculo del flujo óptico fue utilizado el método de Lucas y Kanade [16]. Sin embargo, al trabajar con el Método de Lucas y Kanade, se corre el riesgo de que si se usan ventanas pequeñas y el desplazamiento de los objetos es grande, la correspondencia del pixel podría quedar fuera de la ventana. Para evitar este problema, la implementación del algoritmo de manera piramidal fue implementada, donde se empieza a buscar del más alto nivel de la pirámide de la imagen (menos detallada) hasta los niveles más bajos (detalles más finos).

En la Figura 3 son presentados los vectores de flujo óptico. Puede observarse que los vectores creados a partir de los puntos de interés sobre del vehículo, son más grandes en magnitud, puesto que el vehículo presenta un movimiento mayor al de todos los demás objetos que están en la escena.



Fig. 3. Vectores del flujo óptico

4.3. Filtrado de vectores de flujo óptico

Cuando el flujo óptico es calculado, para obtener mayor precisión suele ser procesado a nivel subpixel, lo que suele llevar a la pérdida de información cuando se quiere trabajar a nivel de pixeles posteriormente. Si el desplazamiento ocurrió en menos de un pixel en las dos direcciones, la magnitud del vector será cero. En la segmentación es requerido que por lo menos haya habido un movimiento considerable, por lo que todos los vectores cuya magnitud sea menor a un umbral β_1 , son eliminados. También puede darse el caso de tener vectores muy grandes cuando se calcula el flujo óptico en algunas texturas como las de sal y pimienta [21] (por ejemplo las texturas de pasto y hojas de árboles). Aunque sean buenos puntos de interés, pueden existir varias correspondencias al mismo punto, lo que traería como consecuencia el encontrar desplazamientos muy grandes. Por lo que también deben ser eliminados los vectores cuya magnitud superen un umbral β_2 . Este filtrado es realizado con la finalidad de evitar ruido. En la Figura 4 son mostrados los vectores de flujo óptico después del filtrado, es decir, solo son mostrados los vectores cuya magnitud están en el intervalo (β_1, β_2) , por lo que se concluye que son los vectores que representan el movimiento de los objetos.



Fig. 4. Filtrado de vectores del flujo óptico con $\beta_1 = 1$ y $\beta_2 = 15$.

4.4. Creación de regiones

Para crear cada una de las regiones que definen a los objetos, es necesario agrupar los vectores del flujo óptico con base en 3 criterios: cercanía, dirección y magnitud.

A y B se consideran vectores cercanos si los puntos donde inician los vectores (puntos de interés) no difieren tanto en x como en y de un umbral γ , es decir:

$$\text{abs}(A(x) - B(x)) < \gamma \text{ or } \text{abs}(A(y) - B(y)) < \gamma \quad (1)$$

A y B se consideran vectores de dirección similar si el ángulo entre esos dos vectores no sobrepasa un umbral δ , es decir:

$$\text{ángulo}(A, B) < \delta \quad (2)$$

A y B se consideran vectores similares en cuanto a su magnitud, si la diferencia de sus normas no difiere de un umbral ε , es decir:

$$\text{abs}(\|A\| - \|B\|) < \varepsilon \quad (3)$$

Con las tres condiciones, son agrupados los vectores de la imagen A en 3 regiones diferenciadas por color mostradas en la Figura 5:

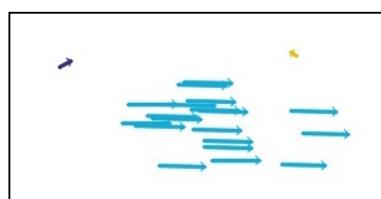


Fig. 5. Imagen que muestra la creación de 3 regiones dadas las condiciones de agrupación con $\gamma = 20$, $\delta = 25$ Y $\varepsilon = 10$.

4.5. Obtención del polígono descriptivo del objeto

Una vez agrupado los vectores en regiones, es posible obtener un polígono que represente la cubierta convexa de la nube de puntos de una región. Tomando en cuenta la ecuación de la recta que pasa por dos puntos:

$$y - y_1 = ((y_2 - y_1)/(x_2 - x_1)) (x - x_1) \quad (4)$$

Es posible obtener la siguiente ecuación donde los términos independientes quedan de un solo lado de la igualdad como es mostrado en la ecuación (5).

$$(y_2 - y_1)x - (x_2 - x_1)y = (y_2 - y_1)x_1 - (x_2 - x_1)y_1 \quad (5)$$

La ecuación 5 por lo tanto representa la condición para que un punto (x, y) pertenezca a la recta formada por los puntos (x_1, y_1) y (x_2, y_2) . Por lo tanto podemos representar la condición para saber si dos puntos pertenecen a la cubierta convexa del conjunto de puntos S. Dos puntos $P(x_1, y_1)$ y $Q(x_2, y_2)$ pertenecen al conjunto de la cubierta convexa C si y solo si, todos los puntos $R(x_3, y_3)$ que pertenezca a S (excepto P y Q) al ser evaluados en la ecuación de la recta que pasa por los puntos P y Q, están en un solo lado de la recta (el signo debe tener un solo valor para todo punto R, ya sea positivo o negativo). Es decir, se cumple la ecuación (6) o (7), pero no ambas.

$$P(x_1, y_1) \in C \text{ y } Q(x_2, y_2) \in C \leftrightarrow \forall R(x_3, y_3) \in S \mid (y_2 - y_1)x_3 - (x_2 - x_1)y_3 < (y_2 - y_1)x_1 - (x_2 - x_1)y_1, P \neq Q \neq R \quad (6)$$

$$P(x_1, y_1) \in C \text{ y } Q(x_2, y_2) \in C \leftrightarrow \forall R(x_3, y_3) \in S \mid (y_2 - y_1)x_3 - (x_2 - x_1)y_3 > (y_2 - y_1)x_1 - (x_2 - x_1)y_1, P \neq Q \neq R \quad (7)$$

Esta manera de calcular la cubierta convexa es de orden n^3 , donde n es el número de puntos de la región. Cabe mencionar que para poder identificar un polígono, es necesario que la región tenga por lo menos 3 puntos para cerrar el polígono. En la Figura 6 es mostrado en color rojo el polígono descriptivo del objeto que es delimitado por la cubierta convexa de los puntos de interés que describen el movimiento del automóvil. Dicho polígono está definido por el inicio de los vectores del flujo óptico

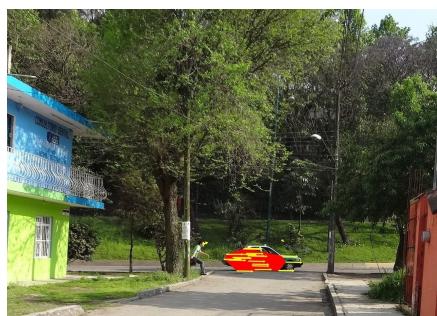


Fig. 6. Polígono formado por la cubierta convexa del automóvil en movimiento.

4.6. Formación de un blob por similitud de color

La intención de este paso, es formar un blob (agrupación de píxeles conexos con una forma irregular) que se adapte de mejor manera a la forma del objeto. Partiendo de que los puntos que forman el área del polígono que describe el movimiento del objeto forman también parte del objeto mismo, es posible encontrar los píxeles conexos con color similar mediante un crecimiento de regiones utilizando el algoritmo “Flood fill”. Flood fill (llenado por inundación) [8][20] es una función que a menudo es usada para marcar o aislar partes de una imagen para su procesamiento o análisis. Consiste básicamente en tomar una semilla (el color de un pixel) e ir revisando iterativamente si por lo menos un vecino de ese pixel tiene color similar. Si lo anterior ocurre entonces esos píxeles similares se agregan a la región de la semilla. Este proceso continúa hasta no encontrar al menos un vecino que tenga color similar. Para este trabajo, las semillas para el algoritmo son todos los píxeles que están dentro del polígono descriptivo del objeto. Con este proceso son agrupados todos los píxeles de un objeto, aunque sean de diferente color entre sí, pero similar a cualquiera de los puntos del polígono. Para realizar este proceso, deben ser asignados dos umbrales τ_1 y τ_2 que definen los límites inferior y superior de discrepancia entre el valor cromático del pixel dentro del polígono y de su vecino. En la Figura 7 es presentada la imagen con la aplicación del algoritmo Flood Fill (llenado por inundación) tomando como semillas todos los píxeles del polígono descriptivo del automóvil tomando como parámetros $\tau_1 = 9$ y $\tau_2 = 9$.

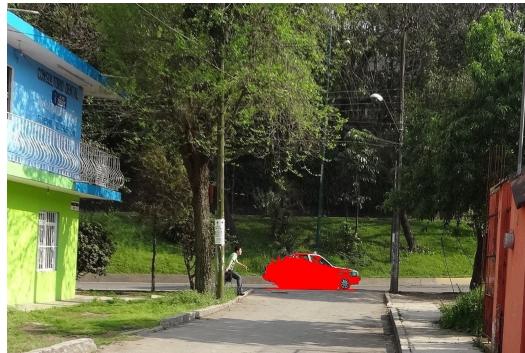


Fig. 7. Resultado de la aplicación de la operación Flood Fill.

5. Resultados

A continuación son mostrados otros resultados del proceso descrito en el artículo. Los parámetros con los cuales fueron obtenidos los mejores resultados (incluyendo los mostrados a continuación) fueron: $\alpha = 0.01$, $\beta_1 = 1$, $\beta_2 = 15$, $\gamma = 20$, $\delta = 25$, $\varepsilon = 10$, $\tau_1 = 9$ y $\tau_2 = 9$. En la Figura 8 es presentado otro ejemplo de la metodología presentada.

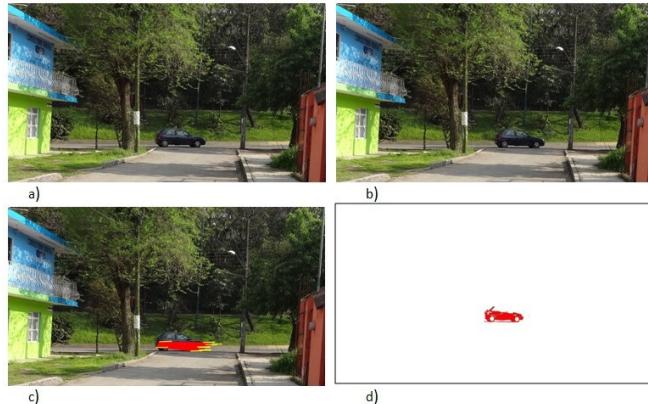


Fig. 8. Las imágenes a y b las capturadas por la cámara, c muestra el polígono con los vectores del flujo óptico y d es la segmentación por color del objeto en movimiento.

6. Conclusiones

La segmentación de objetos por flujo óptico suele ser ambigua debido a diferentes circunstancias como texturas u occlusiones. Con esta propuesta, fue posible segmentar objetos en movimiento en diferentes ámbitos. La importancia de la propuesta radica en que no es requerida información a priori de los objetos móviles que aparecen en la escena, como su forma, color o textura. La identificación de la región que representa a cada objeto es realizada mediante el cálculo de flujo óptico en puntos de interés, lo que disminuye el procesamiento del cálculo. Por último una manera para definir la forma más precisa para cada objeto es realizada mediante semejanza de color de los pixeles que integran el polígono del objeto. Como trabajo futuro podría ser experimentada la creación de un modelo que describa el movimiento de los objetos segmentados en las escenas para inferir sus propiedades.

Referencias

1. Ayer, S., Sawhney, H.: Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding. ICCV, pp. 777-784 (1995)
2. Ballard, D. H.: Generalizing the Hough transform to detect arbitrary shapes. In: Pattern Recognition, Vol. 13, Issue 2, pp. 111-122 (1981)
3. Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, J., Szeliski, R.: A Database and Evaluation Methodology for Optical Flow. In: International Journal of Computer Vision, Vol. 92, Issue 1, pp. 1-31 (2011).
4. Farrokhina, F., Jain, A.: A multi-Chanel Filtering Approach to Texture Segmentation. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (1991)
5. Fleye, F.: Color Detection and Segmentation for Road and Traffics Signs. In: IEEE Conference on Cybernetics and Intelligent Systems, Singapore (2004)

6. Fuh, C., Cho, S., Essig, K.: Hierarchical Color Image Region Segmentation for Content-Bases Image Retrieval System. In: *IEEE Transactions on Image Processing*, Vol. 9, No. 1 (2000)
7. Harris, C., Stephens, M.: A combined corner and edge detector. *Proceedings of the 4th Alvey Vision Conference*, pp. 147–151 (1988)
8. Heckbert, P.: *A Seed Fill Algorithm (Graphics Gems I)*, New York: Academic Press (1990)
9. Heisele, B., Krebel, U., Ritter, W.: Tracking non-rigid, moving objects based on color cluster flow. In: *CVPR*, pp. 253 - 257 (1997)
10. Horn, B. K. P., Schunck, B. G.: Determining optical flow. In: *Artificial Intelligence* 17, pp. 185- 203 (1981)
11. Jepson, A., Black, M.: Mixture models for optical flow computation. In: *CVPR*, pp. 760- 761 (1993)
12. Khan, S., Shah, M.: Object based segmentation of video using color, motion and spatial information. In: *CVPR*, pp. 746- 751 (2001)
13. Leavers, V. F.: Which Hough Transform?. In: *CVGIP: Image Understanding*. Vol. 58, pp. 250- 264 (1993)
14. Littman, E., Ritter, H.: Adaptive Color Segmentation - a Comparison of Neural and statistical Methods. In: *IEEE Transactions on neural networks*, Vol. 8, No. 1 (1997)
15. Lu, C., Chung, P., Chen, C.: Unsupervised texture segmentation via wavelet transform. In: *Pattern Recognition*, vol. 30, issue 5, pp. 729 - 742 (1997)
16. Lucas, B. D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: *Proceedings of the 1981 DARPA Imaging Understanding Workshop* pp. 121-130 (1981)
17. Mora, D., Páez, A., Quiroga, J.: Detección de objetos Móviles en una Escena utilizando Flujo Óptico. In: *XIV Simposio de Tratamiento de Señales, Imágenes y Visión Artificial, Colombia* (2009)
18. Mukherjee, D., Deng, Y., Mitra, S.: Region based video coder using edge flow segmentation and hierarchical affine region matching. In: *SPIE, Visual Communications and Image processing*, vol. 3309, pp. 338-49 (1998)
19. Nagel, H., Steps towards a cognitive vision System. IAKS Karlsruhe.
20. Shaw, R.: QuickFill: An efficient flood fill algorithm. <http://www.codeproject.com/gdi/QuickFill.asp>.
21. Shi, J., Tomasi, C.: Good features to track. In: *9th IEEE Conference on Computer Vision and Pattern Recognition* (1994)
22. Tao, H., Sawhney, H., Kumar, R.: A global matching framework for stereo computation. In: *ICCV*, pp. 532-539 (2001)
23. Wang, J., Adelson, E.: Representing moving images with layers. In: *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 625-638 (1994)
24. Weiss, Y., Adelson, E.: A unified mixture framework for motion segmentation: Incorporating spatial coherence and estimating the number of models. In: *CVPR*, pp. 321- 326 (1996)
25. Wu, Y., Liu Q., Huang, S.: An Adaptive Self-Organizing Color Segmentation Algorithm with Application to Robust Real-Time Human Hand Localization. In: *Asian Conf. on Computer Vision, Taiwan* (2000)
26. Zitnick, C. L., Jovic, N., Kang, S. B.: Consistent Segmentation for Optical Flow Estimation. In: *Proceeding ICCV proceedings of the tenth IEEE International Conference on Computer Vision*, Vol. 2, pp. 1308 - 1315 (2005)