

## DETECCIÓN DE TEXTURAS EN IMÁGENES DIGITALES USANDO EL MODELO DE LATTICE BOLTZMANN

Virginia Cifuentes<sup>a,b</sup>, Mariana del Fresno<sup>a,b</sup>, Alejandro Clause<sup>a,c</sup>

<sup>a</sup> PLADEMA, Fac. Ciencias Exactas, Universidad Nacional del Centro, Tandil, Argentina

<sup>b</sup> CIC, Provincia de Buenos Aires, Argentina

<sup>c</sup> CNEA, CONICET, Argentina

{cifuyente, mdelfres, clause}@exa.unicen.edu.ar, <http://www.pladema.net/>

**Palabras Clave:** Autómatas de Lattice Boltzmann, Detección de Texturas, Segmentación de imágenes.

**Resumen.** La detección de texturas es una importante técnica que facilita la detección de regiones dentro de una imagen con propiedades de textura significativas. Este proceso tiene potenciales aplicaciones en la mejora de imágenes, clasificación de objetos, computación gráfica, visión computacional y especialmente segmentación de imágenes. Aunque hay numerosas técnicas disponibles, no todas son capaces de detectar adecuadamente las texturas, debido a algunas limitaciones inherentes. Este trabajo presenta la aplicación de autómatas de Lattice- Boltzmann para la detección de regiones texturadas en imágenes digitales. La segmentación producida y la eficiencia de la clasificación del método propuesto resultan adecuadas y se obtiene en tiempos razonables.

## 1 INTRODUCCIÓN

Se han propuesto diversos algoritmos de segmentación, los cuales permiten dividir una imagen en sus estructuras de interés. Esta tarea, que no suele requerir mayor esfuerzo para el sistema de visión humano, generalmente se convierte en un desafío considerable al desarrollar algoritmos computacionales, especialmente si se trata de imágenes con estructuras complejas.

Típicamente, la segmentación de imágenes se logra mediante la identificación de los puntos que pertenecen a cada uno de los objetos de interés o a través de la localización de aquellos que determinan su contorno (Pham et al., 2000). Entre las distintas alternativas planteadas, existen algoritmos de segmentación que son capaces de dividir una imagen en regiones uniformes de color o textura. La detección de texturas es una importante técnica que facilita la segmentación de regiones con propiedades de textura significativas. Comúnmente, la textura describe la existencia de alguna regularidad en una región continua de la imagen, la cual se puede manifestar como un patrón o forma repetitivos, aunque no está definido cuán regular debe ser (Pratt, 2001). Este proceso resulta de gran importancia en numerosas aplicaciones de procesamiento de imágenes, especialmente clasificación de objetos, síntesis de texturas para computación gráfica y visión computacional (Amiaz et al., 2007; Fauzi et al., 2006; Sonka, 2007; Zhang et al., 2005).

Entre las distintas técnicas usadas para segmentación de texturas se pueden mencionar los campos aleatorios de Markov, dimensión fractal, polígonos de Voronoi, filtros de Gabor y descomposición wavelet (Kato et al., 2001 y 2006; Peli, 1990; Liu et al., 1997; Keller, 1989; Dennis et al., 1989; Tuceryan y Jain, 1990; Kruizinga et al., 2002). Los tres primeros extraen las características de textura por medio del cálculo de algunos parámetros de la imagen, mientras que los dos últimos primero transforman la imagen al dominio de frecuencia y luego lo dividen en diferentes sub-bandas. La distribución de energía en cada una de estas sub-bandas es utilizada para distinguir diferentes texturas. El principal problema en la mayoría de los métodos se relaciona a la discriminación entre estructuras con diferente textura. Además, la similitud de texturas no puede ser exactamente definida ya que depende por lo general de cada aplicación particular.

A pesar de la amplia variedad de enfoques disponibles, no existe actualmente un esquema que resulte más efectivo para la diferenciación de texturas y la ampliación del conjunto de posibilidades al respecto es aún un problema abierto. En este sentido, este trabajo presenta un algoritmo novedoso de segmentación de imágenes basado en autómatas celulares. El algoritmo es capaz de detectar distintas texturas presentes en una imagen digitalizada analizando magnitudes y parámetros asociados al comportamiento de un fluido simulado mediante el método de Lattice Boltzmann. El algoritmo fue evaluado con imágenes digitales que representan texturas sintéticas y con fotografías reales capturadas de la naturaleza. Los resultados experimentales muestran la efectividad del algoritmo de segmentación basado en el método de Lattice Boltzmann.

## 2 EL MODELO DE LATTICE BOLTZMANN

Los autómatas de Lattice Boltzmann (LB) son aproximaciones relativamente nuevas que utilizan computación paralela para simular fenómenos de transporte de diferentes tipos (Chen y Doolen, 1998). Desde que se presentó el primer modelo bidimensional para representar ecuaciones de fluidos (Succi, 2001), estos autómatas han atraído la atención como métodos prometedores para resolver una variedad de ecuaciones diferenciales parciales y modelar fenómenos físicos. La ventaja del método LB es la velocidad y facilidad del cálculo diferencial a partir de reglas aritméticas simples, presentándose como una alternativa interesante a los métodos numéricos tradicionales. Por ello, el método LB se está convirtiendo en una ponderosa herramienta de ingeniería.

A diferencia de los métodos clásicos que aproximan con esquemas discretos las ecuaciones diferenciales a derivadas parciales que representan comportamientos macroscópicos, LB construye modelos cinéticos locales que incorporan los mecanismos esenciales de procesos que ocurren microscópica o mesoscópicamente, de manera que las propiedades macroscópicas promediadas obedezcan las ecuaciones de transporte deseadas.

LB es esencialmente una representación discreta del espacio mediante una grilla regular de celdas cuyo estado está caracterizado por una población de "partículas". El estado de las partículas se expresa con una función de distribución  $f_i(x,t)$  (no observable directamente) que representa la cantidad de partículas en la celda  $x$  en el tiempo  $t$  moviéndose con los vectores de velocidad  $e_i$  (tratada como una variable interna). La función distribución cambia de estado según reglas que se ejecutan secuencialmente (en lo posible explícitas), emulando mecanismos de transporte y colisión de partículas. Los observables físicos son variables macroscópicas que se generan a partir de los momentos de  $f_i(x,t)$  respecto de la variable  $e_i$ .

El conjunto de velocidades  $e_i$  está definido por el tipo de grilla. La vecindad de Moore en 2D es la más popular, la cual lleva a la resolución conocida como D2Q9 (dos dimensiones y nueve velocidades):

$$\begin{aligned} \bar{e}_0 &= \begin{pmatrix} 0 \\ 0 \end{pmatrix}; & \bar{e}_1 &= \begin{pmatrix} v \\ 0 \end{pmatrix}; & \bar{e}_2 &= \begin{pmatrix} 0 \\ v \end{pmatrix}; \\ \bar{e}_3 &= \begin{pmatrix} -v \\ 0 \end{pmatrix}; & \bar{e}_4 &= \begin{pmatrix} 0 \\ -v \end{pmatrix}; & \bar{e}_5 &= \begin{pmatrix} v \\ v \end{pmatrix}; \\ \bar{e}_6 &= \begin{pmatrix} v \\ -v \end{pmatrix}; & \bar{e}_7 &= \begin{pmatrix} -v \\ v \end{pmatrix}; & \bar{e}_8 &= \begin{pmatrix} -v \\ -v \end{pmatrix}; \end{aligned} \quad (1)$$

donde  $v = \frac{\Delta x}{\Delta t}$  es una velocidad característica dada por el cociente entre el tamaño de la red y el paso de tiempo asociado a cada cambio del estado del autómata.

Como en cualquier autómata celular, el estado de cada celda de un LB cambia siguiendo un esquema de reglas explícitas que sólo dependen del estado actual de la

celda y sus vecinos. La Ec. 2 muestra el paso de transporte desde el tiempo  $t$  al  $t+1$  para una celda en la posición  $x$ :

$$f_i(\vec{x} + \vec{e}_i \Delta t, t + \Delta t) = f_i(\vec{x}, t) \quad (2)$$

En el paso de colisión, se aplica el operador de colisión a la función de distribución de partículas en cada celda. Este operador es clave para el dinamismo del modelo de LB, ya que modela las colisiones de partículas dentro de la celda. Para aplicar el operador de colisión a una celda dada, debe calcularse la función de distribución en equilibrio  $f_i^{eq}$ .

En el caso de un fluido, la densidad  $\rho$  (Ec. 3) de una celda es la medida del número de partículas en la celda:

$$\rho = \sum_{i=0}^{q-1} f_i \quad (3)$$

La velocidad  $u$  (Ec. 4) de la celda es la velocidad promedio de las partículas en tal celda:

$$u = \frac{1}{\rho} \sum_{i=0}^{q-1} f_i e_i \quad (4)$$

La densidad y la velocidad son usadas para calcular la función de distribución en equilibrio  $f_i^{eq}$ . Esta función representa la colisión entre partículas y se define de manera que a nivel macroscópico se recobre el comportamiento de lo que se está simulando (gas, líquido, etc.). En cada paso temporal, se calcula un nuevo conjunto de valores de  $f_i$  (Ec. 5) usando los valores actuales de  $f_i$  y los valores de equilibrio  $f_i^{eq}$ . La regla clásica de actualización de  $f_i(x, t)$  es la ecuación discreta de Boltzmann (*Chen y Doolen, 1998*):

$$f_i(\vec{x} + \vec{e}_i \Delta t, t + \Delta t) = f_i(\vec{x}, t) + \frac{1}{\tau} (f_i^{eq}(\vec{x}, t) - f_i(\vec{x}, t)) + S_i \quad (5)$$

donde  $S_i$  es un término fuente y  $\tau$  es un tiempo característico de relajación usado para controlar la velocidad con la que la simulación alcanza el equilibrio. Tal coeficiente se corresponde con la viscosidad del fluido simulado, de modo que valores muy altos de  $\tau$  corresponden a viscosidades muy bajas.

La ecuación de Lattice Boltzmann generalmente se resuelve mediante un algoritmo de paso temporal explícito en el que se separa la fase de colisión y la de propagación. Esta característica resulta en un método poco eficiente para el cálculo de configuraciones con solución estacionaria. La implementación propuesta suple tal deficiencia modificando la función de equilibrio para que se logre la solución en tiempos acotados. Finalmente, en el paso de colisión se incluye en la función de equilibrio:

$$f_i^{eq} = \frac{\rho e^{-\frac{\sqrt{|u_i-v_i|}}{c}}}{\left( \sum_{\alpha=0}^{q-1} e^{-\frac{\sqrt{|u_\alpha-v_\alpha|}}{c}} \right)} \quad (6)$$

basada en la entropía máxima, donde  $c = 1/3$ .

### 3 APLICACIÓN DE LB AL PROCESAMIENTO DE IMÁGENES

Se puede aplicar LB a imágenes teniendo en cuenta que una imagen digital es un campo escalar de tonos. Este campo puede ser usado para generar un campo fuente  $S_i(x,y)$  para la Ec. 5, que "guiará" al movimiento del fluido imaginario según la textura local. En este trabajo se consideran imágenes en escala de grises, por lo que se tendrá tonos enteros entre 0 y 255. Se postula una fuente de momento (i.e. fuerza) proporcional al gradiente de intensidad local de la imagen  $G(x,y)$ , es decir:

$$S_i = \begin{cases} 0 & i = 0 \vee \exists i / f_i = 0 \\ -\left( \frac{f_i e_i G(x)}{|G(x)| |e_i|} \right) & \forall i \neq 0 \wedge f_i \neq 0 \end{cases} \quad (7)$$

Además se introduce una fuente de masa constante y uniforme en cada celda. Este escenario lleva a un estado estacionario en el cual todo el fluido que entra en cada celda sale por los bordes.

Inicialmente, la imagen digital se transforma en una superficie 2.5 dimensional rectangular e impermeable. El problema debe ser resuelto en un cierto dominio con las condiciones iniciales y de contorno apropiadas y suficientes. En el caso que nos ocupa, inicialmente se supone un campo LB uniforme en todo el dominio. Para las condiciones de contorno se considera el ingreso de flujo uniforme por los cuatro bordes. El algoritmo evoluciona hasta alcanzar el estado estacionario, calculando las nuevas distribuciones de partículas  $f_i$  a partir de la distribución anterior como lo indica la Ec. (5).

Finalmente, con los valores del estado estacionario se generan campos promedio asociados al comportamiento de un fluido: densidad, velocidad promedio, entropía, energía interna y rotor. Estos campos forman imágenes asociadas, que son posteriormente segmentadas mediante un umbralado simple.

## 4 RESULTADOS

### 4.1 Detección en casos sintéticos

Al centro de la Figura 1 se presenta una imagen digital elaborada a partir de los patrones de textura detallados a su izquierda. Ambos patrones corresponden a

cuadrículas de 3x3 píxeles que mantienen la intensidad promedio, aunque difieren en la distribución de píxeles blancos y negros.

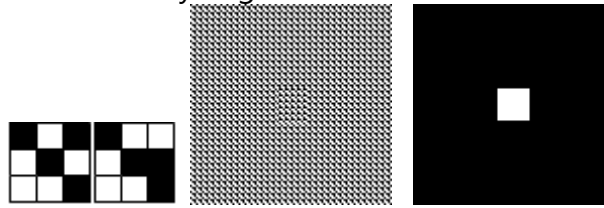


Figura 1. Patrones de textura, imagen digital y segmentada

Se analizaron situaciones de interés práctico para reflejar la dinámica del fenómeno estudiado y la comparación apropiada con los correspondientes resultados analíticos. En los experimentos, se probaron diversas configuraciones del dominio donde las condiciones de contorno y las celdas interiores del autómata actúan como fuentes o sumideros indistintamente.

Primeramente se estudió el flujo sobre la textura simétrica presentada en la Figura 1. Las celdas del autómata se instanciaron con valores iniciales de  $f_i = 0.05$ , la fuente de masa en cada celda es  $s_i = 0.5$  y el parámetro  $\tau$  se fijó en 0.01. Sobre los campos promedios de entropía (H), densidad ( $\rho$ ), módulo de la velocidad ( $u$ ), energía interna (E) y módulo del rotor (R) se aplicó un filtro de umbralado generando imágenes binarias que representan las segmentaciones del objeto "texturado" en el centro de la imagen, rotado con respecto al resto de la imagen (de manera que cambia la textura central).

A fin de evaluar los resultados, se consideró además una segmentación de referencia (a la derecha en Figura 1) y se aplicaron métricas de comparación entre ésta y la segmentación obtenida mediante el algoritmo, las cuales han sido utilizadas en distintos trabajos como indicadores cuantitativos de calidad de la segmentación (del Fresno et al, 2009):

$$CP = \frac{|V_i \cap V_a|}{|V_i \cup V_a|}; FNP = \frac{|V_i - V_a|}{|V_i \cup V_a|}; FPP = \frac{|V_a - V_i|}{|V_i \cup V_a|} \quad (8)$$

$V_i$  y  $V_a$  son los conjuntos de puntos clasificados como objeto en la imagen de referencia y por el algoritmo, respectivamente.  $CP$  es la proporción de coincidencias entre la segmentación ideal ( $V_i$ ) y la obtenida por el algoritmo ( $V_a$ ) respecto de la unión de ambos grupos de puntos.  $FNP$  es la tasa de falsos negativos; es decir, los puntos que corresponden al objeto de interés en la imagen ideal pero no fueron detectados por el algoritmo y  $FPP$  es la tasa de falsos positivos, o sea aquellos puntos que no figuran en la imagen ideal como parte del objeto pero que resultaron clasificados por el algoritmo.  $CP$  alcanza el valor 1 cuando  $V_i$  y  $V_a$  coinciden plenamente y 0 cuando no hay similitud entre los conjuntos segmentados. Por otro lado,  $FNP$  y  $FPP$  producen un valor 0 cuando no hay fallas en la segmentación obtenida mediante el algoritmo con respecto a la segmentación de referencia. La Figura 2 muestra el porcentaje de coincidencias para diferentes campos, de acuerdo a

variaciones del umbral de segmentación, el cual se expresa como un porcentaje del rango de valores de la imagen.

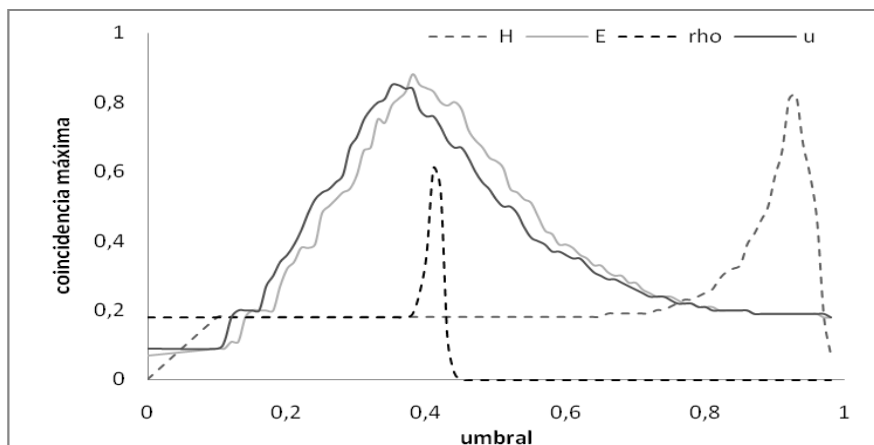


Figura 2. Calidad de la segmentación usando distintos campos fluidos como variable de umbralado.

#### 4.2 Detección en casos reales

También se analizaron imágenes reales con diferencias de textura. En la Figura 3 existen dos tipos de árboles que presentan texturas bien definidas pero de difícil detección automática. A la derecha, la imagen de referencia disgrega la porción representada del pino del resto conformado por los eucaliptus.



Figura 3. Textura de árboles e imagen de referencia.

Se usaron los mismos parámetros de LB que en el caso sintético. La Figura 4 muestra los campos promedios obtenidos en el estado estacionario. En la Figura 5 se muestra un gráfico en el que se pueden apreciar las coincidencias obtenidas con la energía y la densidad. Se puede observar que con el campo de energía pueden distinguirse las dos texturas con una precisión del 97 %, mientras que con el de densidad se logra sólo 64 %.

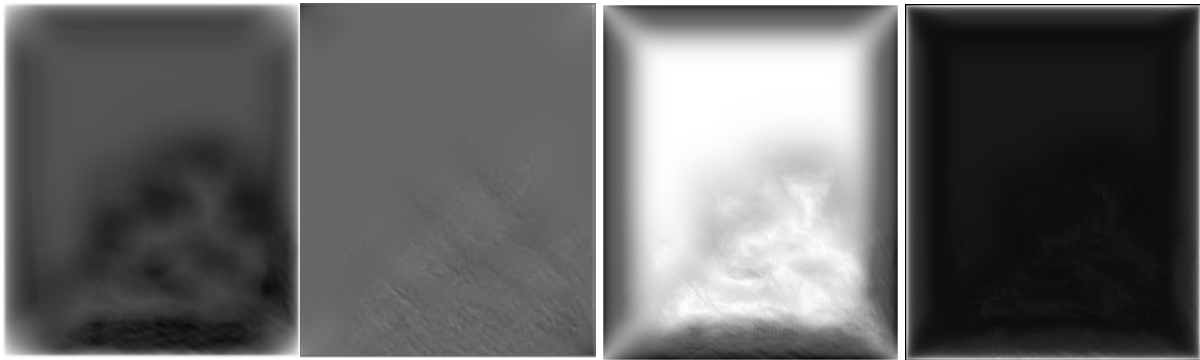


Figura 4. Información simulada: energía interna, rotor, velocidad promedio ( $u$ ) y entropía.

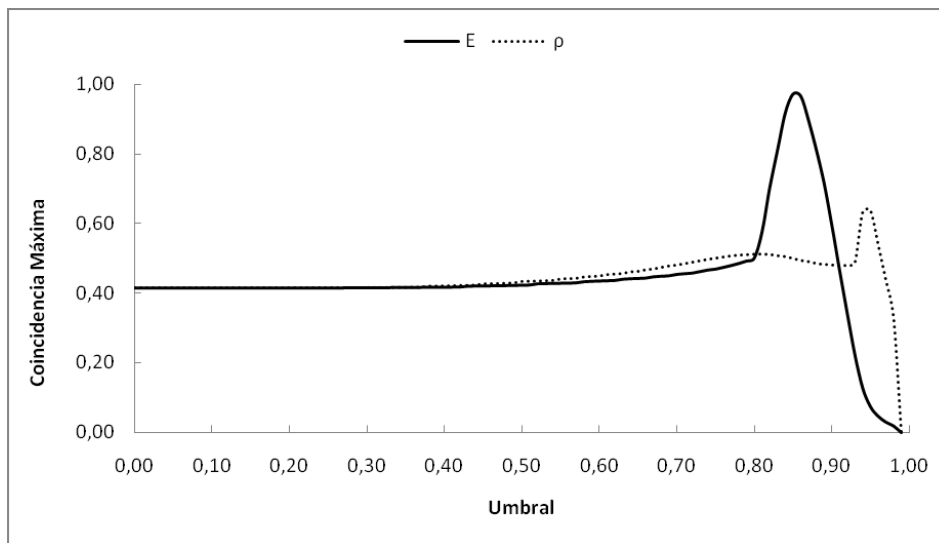


Figura 5. Análisis del flujo sobre texturas de árboles.

## 5 CONCLUSIONES

Se presentó un método novedoso para segmentar automáticamente zonas de diferente textura usando autómatas de Lattice Boltzmann. El concepto básico es hacer circular un "fluido" imaginario sobre la figura, el cual se comportará diferente según la textura local. Los resultados preliminares usando un escenario de fuente interna de masa y bordes salientes son prometedores, logrando segmentaciones que superan ampliamente a las que pueden obtenerse mediante otros métodos tradicionales, por lo que es de esperar que variaciones o extensiones del método permitan alcanzar niveles de calidad mayores.

## REFERENCIAS

Amiaz, T., Fazekas, S., Chetverikov, D. y Kiryati, N., Detecting regions of dynamic texture, *International Conference on Scale Space and Variational Methods in Computer Vision*, LNCS, vol. 4485: 848-859, 2007.



- Chen S. y Doolen G.D., Lattice Boltzmann Method for Fluid Flows. *Annual Reviews Fluid Mechanics*, vol. 30: 329-64, 1998.
- del Fresno M., Vénere M. y Clause A., A combined region growing and deformable model method for extraction of closed surfaces in 3D CT and MRI scans. *Computerized Medical Imaging and Graphics*, vol. 33: 369-376, 2009.
- Dennis, T.J. y Dessiripis, N.G., Fractal modelling in image texture analysis. *IEEE Proceedings of Radar and Signal Processing*, vol. 136 (5): 227-235, ISSN 0956-375X, 1989.
- Fauzi, M. y Lewis, P., Automatic texture segmentation for content-based image retrieval application. *Pattern Analysis and Applications*, vol. 9 (4) : 307-323, ISSN 1433-7541, 2006.
- Kato, Z., Pong, T-Chuen, y Lee, J. Chung-M, Color image segmentation and parameter estimation in a markovian framework. *Pattern Recognition Letters*, vol. 22:309-321, 2001.
- Kato, Z. y Pong, T-Chuen, A markov random field image segmentation model for color textured images. *Image and Vision Computing*, vol. 24:1103-1114, 2006.
- Keller, T., Texture description and segmentation through fractal geometry. *Computer Vision, Graphics and Image Processing*, vol. 45:150-166, 1989.
- Kruizinga, P., Petkov, N. y Grigorescu, S., Comparison of texture features based on Gabor Filters. *International Conference on Image Analysis and Processing*, pp. 100-150. Venecia, Italia, 2002.
- Liu, Y. y Li, Y., Image feature extraction and segmentation using fractal dimension. *International Conference on Information and Signal Processing*, pp. 975-979, IEEE, 2007.
- Peli, T., Multiscale fractal theory and object characterization. *Journal of the Optical Society of America*, vol. 7(6):1113-1123, 1990.
- Pham D., Xu C., Prince J., A survey of current methods in medical image segmentation. *Annual Review of Biomedical Engineering*, 2: 315-37, 2000.
- Pratt W. *Digital Image Processing*, 3<sup>rd</sup> Edition, John Wiley & Sons, Inc. 2001.
- Sonka, M., Hlavac, V. y Boyle, R., Image Processing, Analysis, and Machine Vision. *Thomson-Engineering*, 2007.
- Succi S., The Lattice Boltzmann Equation for Fluid Dynamics and Beyond. *Numerical Mathematics and Scientific Computation*, Clarendon Press, Oxford, 2001.
- Tuceryan, M. y Jain A.K., Texture Segmentation Using Voronoi Polygons. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-12: 211-216, 1990.
- Zhang, S. y Lu, H., Learning texture classifier for flooded region detection in SAR images. *International Conference on Computer Graphics, Image and Vision*, 93-98, 2005.