

# Algoritmo para la detección de zonas poco texturizadas en imágenes reales, para su uso en sistemas de visión estereoscópica

S. Ibarra Delgado, J. Flores Troncoco, H. Gamboa Rosales, R. Soule de Castro

## *Algorithm for detection of low-textured zones in real images used in stereoscopic vision systems*

Recibido: junio 1, 2012  
Aceptado: agosto 5, 2012

Palabras clave: Visión Estereoscópica; técnicas de agregación; medidas de costo.

### **Abstract:**

*The use of stereoscopic vision systems in no-structured real scenes is a common task today. One feature in real images is the presence of low-textured regions with no enough information to select a correct match between two pixels. Local algorithms reduce remarkably his performance in this kind of images. In this work we show an algorithm for detection of low-textured zones, this algorithm avoids to find incorrect matches, the incorrect matches are marked for a posterior process. Results show an improvement in the accuracy on the disparity maps when this algorithm is used.*

**Keywords:** Stereoscopic vision; aggregation techniques; matching cost function.

**E**N las últimas décadas ha existido un desarrollo notable en los sistemas de visión estereoscópica. Se han implementado una gran cantidad de algoritmos, que con bastante precisión, encuentran el mapa de disparidad entre dos imágenes. La gran mayoría de estos algoritmos han sido probados bajo el conjunto de imágenes de prueba de [1], sitio que se ha convertido en referencia principal para la prueba de la calidad de los algoritmos de visión estereoscópica. Este sitio provee un conjunto de imágenes de prueba sintéticas, así como también sus *ground truth*. Sin embargo dentro de los nuevos retos planteados por la comunidad de visión por computadora [2], se plantea el probar los algoritmos de visión estéreo con imágenes reales, ya que éstas presentan un conjunto de retos que no se presentan en las imágenes sintéticas. Entre estos retos se encuentra el manejo de distorsiones radiométricas, zonas de baja textura y la carencia de *ground truth* de los mapas de disparidad.

---

Dentro de los nuevos retos planteados por la comunidad de visión por computadora se encuentran el manejo de distorsiones radiométricas, y zonas de baja textura

---

Aunque han sido desarrollados muchos algoritmos de visión estereoscópica, es posible distinguir una serie de estados comunes entre ellos [3]. Inicialmente por medio de un pre proceso las imágenes son alineadas y rectificadas, lo cual facilita los siguientes procesos. Además un filtro de suavizado es aplicado con el objetivo de reducir el ruido en las imágenes. Posteriormente en un primer estado, se calcula el costo de los empates utilizando una medida de similitud entre un par de píxeles, esta medida es calculada por medio de una función de costo la cual puede utilizar medidas paramétricas o no paramétricas. En el segundo estado las medidas de costo son agregadas sobre una región de soporte. Este estado extrae información del vecindario del píxel de referencia y se agrega esta información en una función que tiene como objetivo encontrar una medida de costo más confiable. En el tercer estado el mapa de disparidad es calculado por medio de un método de optimización local o global. Finalmente, en un cuarto estado el mapa de disparidad es refinado.

Utilizando al marco de referencia anteriormente mencionado, la comunidad de visión por computadora ha propuesto una gran cantidad de algoritmos de correspondencia estereoscópica. Los cuales pueden ser clasificados dentro de dos grandes categorías de acuerdo al método de optimización implementado: los denominados algoritmos locales y los algoritmos globales [2]. Los algoritmos globales presentan mejores resultados en términos de precisión. Desafortunadamente, la cantidad de recursos y tiempo que estos consumen, en la mayoría de los casos no lo hacen adecuados para implementaciones que deban cumplir con las restricciones impuestas por los sistema de tiempo real. Por otro lado los algoritmos locales presentan pobres resul-

tados en relación con su precisión, sin embargo su tiempo de procesamiento es menor. Las investigaciones que se han llevado a cabo recientemente [4,5,6] presentan funciones de costo más robustas y técnicas de agregación adaptables [7,8] que permiten que los algoritmos locales obtengan resultados más precisos, comparables a los alcanzados por los algoritmos globales y que permiten obtener tiempos de procesamiento en tiempo real.

---

La comunidad de visión por computadora ha propuesto una gran cantidad de algoritmos de correspondencia estereoscópica

---

Las técnicas de post procesamiento del mapa de disparidad han mostrado mejorar sustancialmente la precisión de los mismos [5]. Dentro de estas técnicas se encuentran la detección de atípicos, interpolación, ajuste de la discontinuidad de profundidades, refinamiento a nivel sub-píxel, entre otros. En este trabajo se presenta un algoritmo que calcula el mapa de disparidad de un par de imágenes estéreo, utilizando una medida de costo robusta y un método de agregación adaptable. Además el pos proceso de interpolación es llevado a cabo por medio de una selección previa de píxeles fuertes o débiles dependiendo de si se encuentran en una zona texturizada o no texturizada. La principal contribución del presente trabajo es el desarrollo de una medida que permite identificar cuando un píxel se encuentra en una zona de baja textura.

El resto del presente trabajo se encuentra organizado de la siguiente forma. La sección II describe la medida de costo combinada y la técnica de agregación utilizada en este trabajo. La sección III describe la técnica para detección de zonas de baja textura. En la sección IV se exponen los resultados obtenidos. Finalmente en la sección V se dan las conclusiones y se presenta el trabajo futuro.

## MEDIDA DE COSTO, TÉCNICA DE AGREGACIÓN Y POST PROCESAMIENTO

Cuando son utilizados algoritmos locales para encontrar la correspondencia estéreo, se hace énfasis en utilizar una función de costo y una técnica de agregación apropiada. Esto es mostrado en detalle en los siguientes párrafos.

### Medida de costo

Muchas aplicaciones de visión estéreo se llevan a cabo en escenarios donde las condiciones de luz varían con el tiempo. Es importante seleccionar una función de costo que garantice obtener una buena precisión bajo estas condiciones. Los resultados presentados en [9], muestran que una función de costo combinada presenta mejor rendimiento que las medidas individuales. Dado un pixel  $p = (x, y)$  en la imagen izquierda y un pixel  $pd = (x - d, y)$  en la imagen derecha para un valor de disparidad  $d$ , la medida de costo propuesta es presentada en (1)

$$C_{(p,d)} = \omega \frac{C_{AD}}{N_{AD}} + (1 - \omega) \frac{C_{SCT}}{N_{SCT}}, \quad (1)$$

donde:

$\omega$  Es el factor de peso asignado a cada una de las funciones de costo individuales.

$C_{AD}$  Es el valor absoluto de la diferencia de las intensidades entre el pixel  $p$  y el pixel  $pd$ .

$N_{AD}$  Es la constante de normalización asignada a la medida AD, con un valor de 255, la cual es la máxima diferencia para un valor codificado en 8 bits.

$C_{SCT}$  Es la distancia de Hamming entre las dos cadenas de bits generadas por la Sparce Census Transform (SCT) [10], para los pixeles  $p$  y  $pd$ .

$N_{SCT}$  Es la constante de normalización asignada a la medida SCT, que tiene un valor de 24, la cual es la máxima diferencia en la distancia de Hamming producida cuando se utiliza una ventana de  $9 \times 9$  en la SCT.

Las constantes de normalización tienen el propósito de generar funciones de costo individuales en el rango de 0 - 1 y prevenir que los resultados sean sesgados hacia un componente individual. La ventaja de utilizar la SCT reside en generar un vector de información más pequeño comparado con la transformada Census tradicional, esto sólo produce una pequeña pérdida en la precisión [11].

### Técnica de agregación

Basado en el hecho de que la zona de agregación solamente debe de contener pixeles que pertenecen al mismo plano de disparidad, existen dos técnicas de agregación que obtienen buenos resultados comparadas con la técnica de ventana fija. Una técnica llamada *Adaptive Support Weight* (ASW) [12], construye una ventana fija y asigna pesos de soporte a cada elemento individualmente. El proceso de agregación es realizado tomando en cuenta estos pesos. La otra técnica llamada *Adaptive Window Shape* (AWS) se basa en ajustar la forma de la ventana utilizando solo pixeles que pertenecen al mismo espacio de disparidad. En este trabajo se aplica la técnica AWS, empleando una cruz de soporte para el proceso de agregación como es sugerido en [7].

El proceso de agregación trabaja de la siguiente forma. Dado un pixel  $p$  (pixel de referencia), se construyen cuatro brazos de longitud variable, tanto en la dirección horizontal  $(h_p^-, h_p^+)$ , como en la dirección vertical  $(v_p^-, v_p^+)$  como se muestra en la Fig. 1. Cada brazo termina cuando uno de los siguientes criterios se cumple:

- La diferencia de intensidades entre el pixel de referencia y el pixel en el brazo excede un umbral predeterminado.
- La longitud del brazo alcanza un valor máximo predefinido.

Para construir la región de soporte  $U(p)$  para el pixel  $p$  se integran múltiples segmentos  $H(q)$  sobre el segmento vertical  $V(p)$  del pixel  $p$ , donde  $q$  es un pixel de soporte localizado en el segmento vertical  $V(p)$ .  $H(q)$  provee un conjunto de pixeles de soporte válidos en el vecindario de  $H(q)$ . Las regiones de soporte son calculadas para el pixel  $p = (x, y)$  en

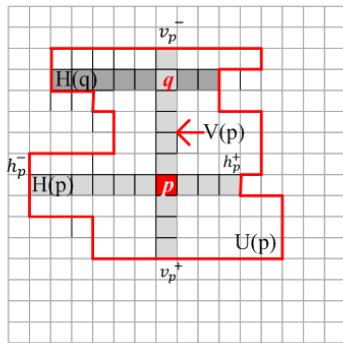


Fig. 1 Construcción de Cruz de Soporte

la imagen derecha y para el píxel  $p = (x - d, y)$  en la imagen derecha. La región de soporte final es calculada con la intersección de las dos regiones de soporte calculadas.

### Post procesamiento

Habiendo sido calculado el mapa inicial de disparidad, es común efectuar una serie de post procesos que tienen como objetivo refinar el mapa de disparidad. Uno de los post procesos que mayor impacto tienen en la precisión del mapa de disparidad es la interpolación. Para esto primero se debe pasar por el proceso de verificación izquierda-derecha (left right check validation, LRC) que tiene como objetivo encontrar las disparidades que fueron mal calculadas.

La verificación LRC parte de los mapas de disparidad calculados derecha-izquierda e izquierda-derecha y se considera que una disparidad en la imagen izquierda está correctamente calculada si se cumple con el criterio:

Si para una disparidad dada en el mapa de disparidad izquierdo  $d_l = D_l(x, y)$ , su correspondiente en el mapa de disparidad derecho  $d_d = D_d(x + d_l, y)$  son iguales, entonces se considera que esa disparidad está correctamente calculada. En caso contrario, la disparidad es descartada y esa posición es marcada como falsa de tal modo que deberá ser tomada en cuenta para el posterior proceso de interpolación. Los píxeles que no pasan la verificación LRC son interpolados utilizando el método de interpolación lineal.

El algoritmo completo que es utilizado en este trabajo primero filtra las imágenes originales [13] que ya han sido rectificadas, y les aplica el filtro de la mediana con el obje-

tivo de suavizar las imágenes y quitar valores atípicos. Posteriormente para cada píxel en las imágenes suavizadas, se calcula la función de costo combinada AD+Census que es la entrada para el proceso de agregación adaptable. Una vez calculadas las funciones de costo agregadas se procede a encontrar el mínimo dentro de todas las disparidades para el píxel de referencia basado en la bien conocida técnica *Winner Takes All* (WTA). Finalmente se efectúa un proceso de refinamiento basado en la interpolación lineal.

### ZONAS DE BAJA TEXTURA

El proceso de interpolación lineal trabaja sobre las posiciones  $(x, y)$  que no cumplen con el criterio LRC. Para estas posiciones se encuentran tanto a la derecha como a la izquierda de la posición en conflicto disparidades que hayan sido correctamente calculadas según la misma LRC. Sin embargo este criterio presenta problemas cuando se tienen zonas de baja textura, lo cual es muy común en imágenes reales. En la Fig. 2 a) se muestra una imagen con una zona poco texturizada, Las Fig. 2 b) y 2 c) muestran el mapa de disparidad izquierda-derecha y el mapa de disparidad derecha-izquierda. Se puede observar que en la zona de baja textura los dos mapas presentan fallos significativos.

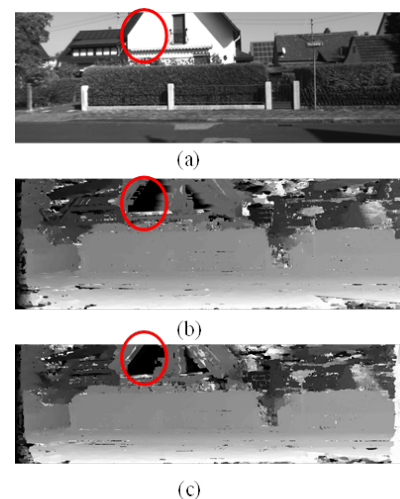


Fig. 2  
 (a) Imagen original con zona de baja textura  
 (b) Mapa de disparidad Izquierda-Derecha  
 (c) Mapa de disparidad Derecha-Izquierda

Al utilizar el criterio de verificación LRC sobre los mapas de disparidad de la Fig. 2, se encuentra que para un conjunto significativo de píxeles dentro de la zona mencionada existe una disparidad de 0 (píxeles que se encuentran en el infinito), lo cual es notoriamente incorrecto. Esto sucede tanto en el mapa izquierda-derecha como en el mapa derecha-izquierda. Lo anterior debido a que en esta zona no hay suficiente información para identificar de una disparidad a otra, entonces para varias disparidades es calculado el mismo mínimo y el criterio WTA toma el primero que encuentra, que generalmente cae en la posición 0. Estas disparidades son consideradas como falsos positivos entonces al momento de realizar la verificación LRC, existen una cantidad significativa de valores que son considerados correctos y que son tomados por el proceso de interpolación como disparidades correctas, generando un cálculo de interpolación erróneo.

El problema que se presenta en la Fig. 2 tiene que ver con la cantidad de información con que se encuentra para realizar un empate, de aquí que sea común el proponer como alternativa de solución aumentar la región de agregación. Con lo anterior se presenta una mejora significativa en los mapas de disparidad calculados, sin embargo esto se logra a cambio de una gran cantidad de recursos utilizados y un tiempo de cómputo alto, lo cual no es aceptable para implementaciones de tiempo real.

Nosotros proponemos una alternativa de solución para evitar obtener falsos positivos y que estos no sean tomados en cuenta como píxeles de soporte en el proceso de interpolación. La idea se basa en que dado un píxel  $p = (x, y)$  determinar si este se encuentra en una zona de baja textura, si es así, marcar esta posición en el mapa de disparidad como falsa, ya que la probabilidad de que éste sea un falso positivo es muy grande. Ahora el problema se convierte en encontrar un método que permita detectar zonas de baja textura.

Según [3] un píxel se encuentra en una zona de baja textura cuando el cuadrado del gradiente horizontal promediado sobre una ventana cuadrada se encuentra debajo de un umbral predeterminado. Al aplicar este criterio se encuentra un mapa de zonas de baja textura como se muestra en la Fig. 3 a). Sin embargo se puede observar que bajo este criterio existen zonas de baja textura que en el mapa de disparidad

se encuentran correctamente calculadas Fig. 3 b), y que si son marcados como falsos positivos, se perderían zonas que fueron correctamente calculadas.

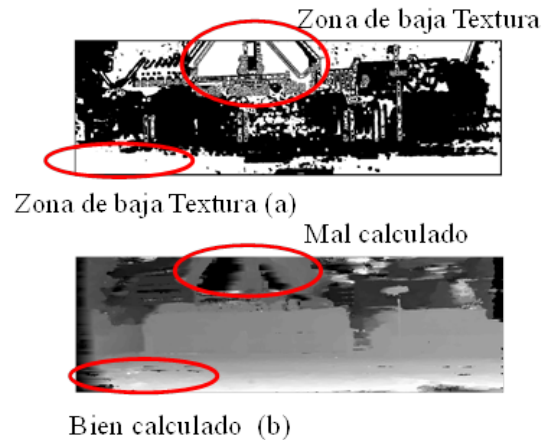


Fig. 3  
 (a) Mapa con zonas de baja textura (blanco)  
 Calculado con [3]  
 (b) Mapa de disparidad

Proponemos una alternativa de solución para evitar obtener falsos positivos y que estos no sean tomados en cuenta como píxeles de soporte en el proceso de interpolación

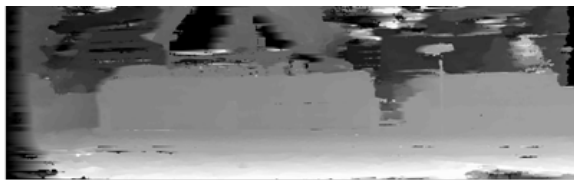
La técnica de agregación que es utilizada en este trabajo [7], delimita la zona de agregación por medio de un umbral de diferencia de intensidad. Esto da una idea de cuan texturizado está el vecindario del píxel ancla. De tal modo que nosotros proponemos un criterio en el que se decide si un píxel se encuentra en una zona texturizada basada en la zona de agregación encontrada, de acuerdo al siguiente criterio: Un píxel se considera que se encuentra en una zona texturizada, si la suma del valor absoluto del píxel ancla, con todos los píxeles que se encuentran en la zona de agregación supera un umbral establecido. Bajo este criterio se



Tabla 1. Porcentaje de pixeles que pasan la verificación LRC.

Peso $\omega$	Sin criterio de baja textura	Con criterio de baja textura
0.1	% 66.43	% 75.57
0.3	% 67.28	% 76.05
0.5	% 67.87	% 76.32
0.7	% 67.51	% 76.56
0.9	% 62.36	% 76.67

probó para diferentes valores, encontrando que un valor de umbral de 100 puntos en diferencia de intensidad, encuentra zonas de baja textura solo en las zonas que tienen falsos positivos. La Fig. 4. Muestra que se encuentra una zona de baja textura solamente en regiones que contienen falsos positivos, de tal modo que esta región es etiquetada como falso para el posterior proceso de interpolación.



(a)



(b)

Fig. 4

- (a) Mapa de disparidad
- (b) Zona de baja textura con criterio propuesto

## RESULTADOS

El proceso de interpolación lineal es realizado tomando en cuenta las zonas de baja textura encontradas en el proceso detallado anteriormente. En la Fig. 5 se muestra el mapa

de disparidad generado después de aplicado este proceso, en ésta se puede observar que la zona que anteriormente presentaba falsos positivos es interpolada correctamente y que el mapa de disparidad presenta mejores resultados.



Fig. 5

Mapa de disparidad interpolado con el criterio de propuesto de baja textura

Para evaluar la medida de costo y la efectividad de la técnica propuesta se procedió a variar el peso  $\omega$  y se calculó la precisión del mapa de disparidad tomando como criterio de empate correcto una diferencia de 2 puntos en la disparidad entre el mapa izquierda-derecha y el mapa derecha-izquierda bajo el criterio LRC. En la Tabla 1 se muestran los resultados, en todos los casos se puede observar que existe una mejora significativa al aplicar el criterio propuesto.

## CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se presentó una técnica de detección de zonas de baja textura que permite refinar los mapas de disparidad con mayor precisión, los resultados muestran una

mejora significativa al utilizar la técnica propuesta. En el trabajo futuro se investigará como encontrar un criterio para zonas de baja textura para otro tipo de técnicas de agregación como es la ASW.

## Bibliografía

- [1] <http://vision.middlebury.edu/stereo/>, 2013.
- [2] <http://hci.iwr.uni-heidelberg.de/Static/eccv2012/>
- [3] D. Scharstein and R. Szeliski, “A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms,” *IJCV*, vol. 47, no. 13, pp. 7-42, Apr. 2002.
- [4] N. Einecke and J. Eggert, “A Two-Stage Correlation Method for Stereoscopic Depth Estimation”, *Digital Image Computing: International Conference on Techniques and Applications (DICTA) 2010*, pp. 227-234, 1-3 Dec. 2010.
- [5] K. Ambrosch and W. Kubinger, “Accurate hardware-based stereo vision”, *Computer Vision and Image Understanding*, Volume 114, Issue 11, Pages 1303-1316, November 2010.
- [6] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang and X. Zhang, ”On building an accurate stereo matching system on graphics hardware” *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp.467,474, 6-13 Nov. 2011.
- [7] K. Zhang, J. Lu and G. Lafruit, “Cross-Based Local Stereo Matching Using Orthogonal Integral Images”, *IEEE Transactions on Circuits and Systems for Video Technology*, vol.19, no.7, pp. 1073-1079, 2009.
- [8] K. Yoon and I. Kweon, “Adaptive support-weight approach for correspondence search”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.28, no.4, pp. 650-656, April 2006.
- [9] L. De-Maeztu, A. Villanueva and R. Cabeza, “Stereo matching using gradient similarity and locally adaptive support-weight”, *Pattern Recognition Letters*, Volume 32, Issue 13, Pages 1643-1651, 2011.
- [10] K.-J. Yoon, and I.-S. Kweon, “Locally adaptive support-weight approach for visual correspondence search”. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. 924–931, 2005.
- [11] B. Cyganek, “Matching of the multi-channel images with improved nonparametric transformations and weighted binary distance measures”, *IWCIA’06 Proceedings of the 11th international conference on Combinatorial Image Analysis*, Pages 74-88, 2006.
- [12] K.-J. Yoon, and I.-S. Kweon, “Locally adaptive support-weight approach for visual correspondence search”. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. 924–931, 2005.
- [13] <http://www.cvlibs.net/datasets/kitti/index.php>, “The KITTI Vision Benchmark Suite”.

## Acerca del autor o autores

Salvador Ibarra Delgado es Ingeniero en Sistemas Electrónicos, egresado del Instituto Tecnológico y de Estudios Superiores de Monterrey en 1986. Obtuvo el grado de Maestro en Informática y Tecnologías Computacionales por parte de la Universidad Autónoma de Aguascalientes en el año 2006. Actualmente se encuentra estudiando el Doctorado en Ingeniería con especialidad en Arquitectura de Computadoras en la Universidad de Córdoba, España. Trabaja actualmente en la Universidad Autónoma de Zacatecas. Sus áreas de interés: Dispositivos Lógicos Programables, Arquitectura de Computadoras, Visión por Computadora.