

PENDETEKSIAN TINGKAT KABUR PADA GAMBAR MENGUNAKAN WAVELET TRANSFORM

Liliana, Kartika Gunadi, Denny Pujo

Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra
lilian@petra.ac.id

Abstract

Taking pictures using digital camera often produces images blurred. Unfocussed lens and object movement when capture the photo, cause the blur effect. Blur is often not able to be clearly perceived by human eye. It takes a good detection of image blurring an important area today. This detection is driven by the need to produce satisfy images with good quality. The detection level of blur is also useful to provide guidance if it will be manipulated using photo manipulation that aims to reduce the blur effect.

In this research, we try to make an application to the detection of blurred images by using wavelet. This detection process begins with Haar Wavelet, continue with counting Emap and construct local maxima. Last step is calculating blur of the image using the local maxima calculation.

The test results showed that the blurred image can be detected and quantified the blur level. In addition to the lens and motion or hand shaking, light intensity and composition of the image itself also affect the level of blur. This is due to the calculation of the number of edges of objects in this method.

Keywords: blur, blur detection, Wavelet Transform, Haar Wavelet Transform, construct emap, local maxima.

1. PENDAHULUAN

Kemudahan penggunaan kamera digital tidak selalu diikuti dengan kemampuan fotografi yang cukup baik sehingga foto yang dihasilkan tidak maksimal. Salah satunya adalah terjadinya blur. Beberapa penyebab blur adalah ketidakfokusan lensa, obyek foto yang bergerak, atau guncangan pada tangan pada saat pengambilan gambar [1],[3],[5].

Untuk fotografi amatir, kerusakan akibat blur ini bisa dimanipulasi dengan piranti khusus

untuk mengedit foto. Piranti semacam ini akan memberlakukan perbaikan blur dengan perlakuan yang sama tanpa memperhatikan tingkat blur pada foto. Hal ini akan menyebabkan hasil perbaikan malah semakin merusak. Foto dengan blur yang kecil sebaiknya dimanipulasi secara berbeda dibandingkan dengan foto yang tingkat blurnya lebih besar.

Salah satu contoh metode untuk memperbaiki blur adalah *Digital Signal Processing* (DSP). Akan tetapi, apabila keseluruhan *image* dilakukan pengurangan blur dengan menggunakan DSP, akan ada kemungkinan foto yang telah bagus akan menjadi buruk, karena belum tentu keseluruhan foto tersebut mengalami blur.

Pendeteksian tingkat blur menjadi sangat penting pada saat ini, karena momen-momen penting yang terjadi sering ingin diabadikan, akan tetapi hasil gambar tersebut kabur tanpa disadari. Penelitian pendeksian tingkat blur diawali pada tahun 1994 oleh W. Sweldens yang menemukan *lifting technique*. Kemudian pada tahun 1996, I. Daubechies dan W. Sweldens mengembangkan *Discrete Wavelet Transform*. Berdasarkan hal itu, maka dikembangkanlah aplikasi untuk mendeteksi kekaburan sebuah gambar. Pada penelitian ini, dikembangkan sebuah perangkat lunak untuk mendeteksi kekaburan foto dengan metode *Wavelet transform* yang dikembangkan oleh Hanghang Tong, Mingjing Li dan Changshui Zhang pada tahun 2004.

2. WAVELET TRANSFORM

Wavelet Transform dapat mengubah sebuah nilai dari data signal yang tidak terlalu merusak data aslinya [1]. Sehingga, apabila dikembalikan akan dapat menemui data yang hampir sama dengan data aslinya.

Tabel 1. Data 1D Haar Wavelet

56	40	8	24	48	48	40	16
48	16	48	28	8	-8	0	12
32	38	16	10	8	-8	0	12
35	-3	16	10	8	-8	0	12

Baris pertama pada Tabel 1 adalah sebuah contoh data, dan diasumsikan bahwa data pada tabel tersebut bukanlah sebuah data acak, melainkan data yang memiliki struktur. Diasumsikan bahwa angka-angka tersebut berkorelasi dengan angka disebelahnya, sehingga setiap kali akan di ambil 2 angka, untuk dihitung rata-rata dan selisihnya. Baris kedua adalah hasil perhitungan baris pertama, berisi 4 nilai rata-rata dan 4 nilai selisih dari angka-angka di baris pertama. Kemudian dilakukan hal yang sama pada 4 angka pertama pada baris ke 2, yang menghasilkan 2 nilai rata-rata, 2 nilai selisih, sedangkan 4 angka terakhir tetap. Hal ini dilakukan lagi sehingga menghasilkan 1 nilai rata-rata dari 8 data tersebut, dan 7 nilai selisihnya.

Untuk mendapatkan kembali angka-angka semula, perhitungan yang ada dapat dibalik sehingga data pada baris pertama dapat diperoleh kembali. Dimulai dari baris terakhir, 2 data pada baris ketiga adalah $32 = 35 + (-3)$ dan $38 = 35 - (-3)$. Kemudian, 4 data pertama pada baris ketiga adalah $48 = 32 + (16)$, $16 = 32 - (16)$, $48 = 38 + (10)$ dan terakhir adalah $28 = 38 - (10)$. Data pada baris pertama dapat dihitung dengan cara yang sama.

Transformasi tersebut juga dapat dilakukan pada image yang memiliki 2 dimensi, yang dapat disebut 2D Haar Transform [2].

Tabel 2. Data 2D

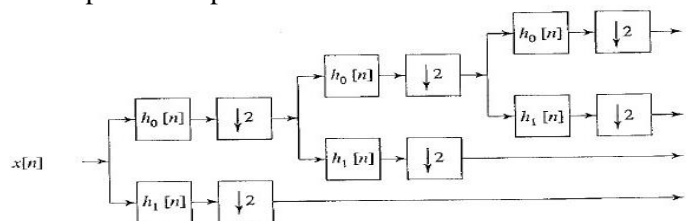
		3	27	27	3		
		27	55	25	27		
				5			
		27	55	55	27		
		3	27	27	3		

Tabel 2 adalah sebuah contoh data untuk Haar Wavelet 2D. Seperti yang dilakukan pada Haar Wavelet 1D, Haar Wavelet 2D adalah mengubah data secara horizontal terlebih dahulu, kemudian secara vertical. Pada contoh tabel 2, 2 angka pertama dan yang terakhir tidaklah terlalu penting karena sama-sama bernilai 0. Dan sisa angka yang ada, dilakukan perhitungan rata-rata dan pengurangan pada angka tersebut. Hasil dari perhitungan tersebut dapat dilihat pada Tabel 3. Pertama-tama proses perhitungan dilakukan secara vertikal, lalu hasil dari perhitungan tersebut dilakukan perhitungan lagi secara horisontal.

Tabel 3. Hasil 2D Haar Wavelet transform dari data pada tabel 2.

0	0	0	0	0	0	0	0
0	143	143	0	0	-48	48	0
0	143	143	0	0	-48	48	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	-48	-48	0	0	16	-16	0
0	48	48	0	0	-16	16	0
0	0	0	0	0	0	0	0

Harr Wavelet Transform adalah langkah awal untuk melakukan Wavelet Transform [1],[3]. Ketika menggunakan Wavelet Transform berulang kali, maka disebut sebagai Discrete Wavelet Transform (DWT). Contoh gambar untuk Discrete Wavelet Transform ini dapat dilihat pada Gambar 1.



Gambar 1. Discrete Wavelet Transform

Pada Gambar 1, $x[n]$ adalah gambar masukan untuk proses DWT. Semua baris pada gambar tersebut diproses hingga menjadi $h_0[n]$ dan $h_1[n]$. $h_0[n]$ yang merupakan rata-rata dari data dan $h_1[n]$ merupakan selisih dari data. Setelah semua baris diproses akan menghasilkan $h_0[n]$ dan $h_1[n]$, bila terdapat jumlah ganjil, maka jumlah ganjil

tersebut dihapus (Ze-nian,2004), dan kemudian $h_0[n]$ dan $h_1[n]$ tersebut digabungkan kembali untuk menghasilkan baris yang sudah tertransformasi. Setelah semua baris sudah ditransformasikan, semua kolom diproses agar menghasilkan $h_0[n]$ dan $h_1[n]$. Bila terdapat jumlah yang ganjil lagi maka jumlah yang ganjil tersebut dihapus. Setelah kedua proses tersebut dijalankan, maka 1 level DWT telah dijalankan, dan akan menghasilkan LL (low-low), HL(high-low), LH(low-high), dan HH(high-high). LL dapat diuraikan kembali untuk melakukan proses pada level selanjutnya. Gambar 2 menunjukkan gambar untuk DWT level 1 dan DWT level 3.

LL	HL	LL ₃	HL ₃	HL ₂	HL ₁ : Horizontal Detail
		LH ₃	HH ₃		
LH	HH	LH ₂	HH ₂	HH ₁ : Diagonal Detail	
		LH ₁ : Vertical Detail			

Gambar 2. DWT level 1 (kiri). DWT level 2 (kanan).

Kemudian perhitungan *Edge Map* nya pada tiap skala digunakan persamaan 1 [4].

$$emap_i(k,l) = \sqrt{LH_i^2 + HL_i^2 + HH_i^2} \quad (i = 1,2,3) \quad (1)$$

Setelah didapatkan *Emap*, maka dilakukan *local maxima* pada tiap window. Ukuran *window* normal untuk skala terbesar adalah 2x2, kemudian yang lebih kasar adalah 4x4, dan yang paling kasar adalah 8x8. Dan hasil nya akan menjadi $Emax_i(i = 1,2,3)$

Dengan *Harr Wavelet Transform*, Emax menunjukkan keintensitasan dari sebuah tepian. Semakin besar Emax, maka semakin besar intensitas tepian tersebut. Dengan *threshold* (batas) yang diberikan, jika $Emax_i(k,l) > threshold$, maka (k,l) merupakan sebuah tepian, jika tidak maka (k,l) bukan merupakan tepian. Dengan melakukan *Harr Wavelet Transform* dalam tipe tepian yang berbeda, maka kita dapat melihat bahwa hasil nya akan berbeda pula. Tabel intensitas untuk tipe-tipe tepian yang berbeda dapat dilihat pada Tabel 4 [3].

Tabel 4. Macam-macam tepian

	Emax 1	Emax 2	Emax 3
Dirac-Structure	Tertinggi	Sedang	Terendah
Astep-Structure	Tertinggi	Sedang	Terendah
Gstep-Structure	Terendah	Sedang	tertinggi
Roof-Structure	Terendah	Sedang	Tertinggi
	Terendah	Tertinggi	Sedang

Dari *Harr Wavelet Transform* tersebut dapat didapatkan beberapa peraturan [4]:

1. Jika $Emax_1(k,l) > threshold$ atau $Emax_2(k,l) > threshold$ atau $Emax_3(k,l) > threshold$ maka (k,l) adalah sebuah tepian dan total dari hasil setiap tepian tersebut dimasukkan kedalam N_{edge}

2. Untuk setiap tepian, jika $Emax_1(k,l) > Emax_2(k,l) > Emax_3(k,l)$ maka tepian tersebut merupakan *Dirac-structure* atau *Astep-structure* dan total dari hasil setiap tepian tersebut dimasukkan kedalam N_{da}

3. Untuk setiap tepian, jika $Emax_1(k,l) < Emax_2(k,l) < Emax_3(k,l)$ maka tepian tersebut merupakan *Roof-structure* atau *Gstep-structure* dan hasil dari total semua tepian tersebut dimasukkan kedalam N_{rg}

4. Untuk setiap tepian, bila $Emax_2(k,l) > Emax_1(k,l)$ dan $Emax_2(k,l) > Emax_3(k,l)$ maka tepian tersebut merupakan *Roof-structure* dan hasil dari total semua tepian tersebut dimasukkan kedalam N_{rg} bersama dengan total dari nomor 3.

5. Untuk Setiap *Gstep-structure* atau *Roof-structure*, jika $Emax_1(k,l) < threshold$ lebih besar kemungkinannya untuk ada dalam *blurred image* dan hasil dari total semua tepian tersebut dimasukkan ke dalam N_{brg}

6. Lalu dilakukan perhitungan perbandingan dari *Dirac-structure* dan *Astep-structure* terhadap keseluruhan *edge* dengan perhitungan

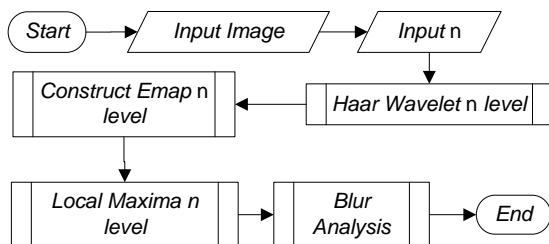
$per = \frac{N_{da}}{N_{edge}}$. jika $per > Minzero$, anggapan image tersebut tidak *blur* dan sebaliknya. Dimana *Minzero* adalah parameter positif yang mendekati nol.

7. Kemudian berapa banyak *Roof-structure* dan *Gstep-structure* yang mengalami efek blur.

$$BlurExtent = \frac{N_{brg}}{N_{rg}}$$

3. GARIS BESAR SISTEM KERJA PERANGKAT LUNAK

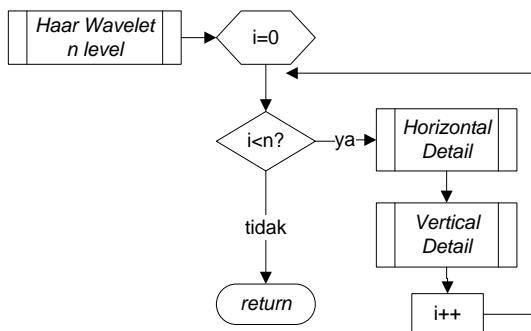
Dalam sistem perangkat lunak untuk pendeteksian *blur* ini, terdapat beberapa tahapan penting yang harus dilakukan. Adapun rancangan sistem kerja perangkat lunak secara garis besar ditunjukkan pada 0.



Gambar 3. Diagram Alir Garis Besar Sistem Kerja Perangkat Lunak

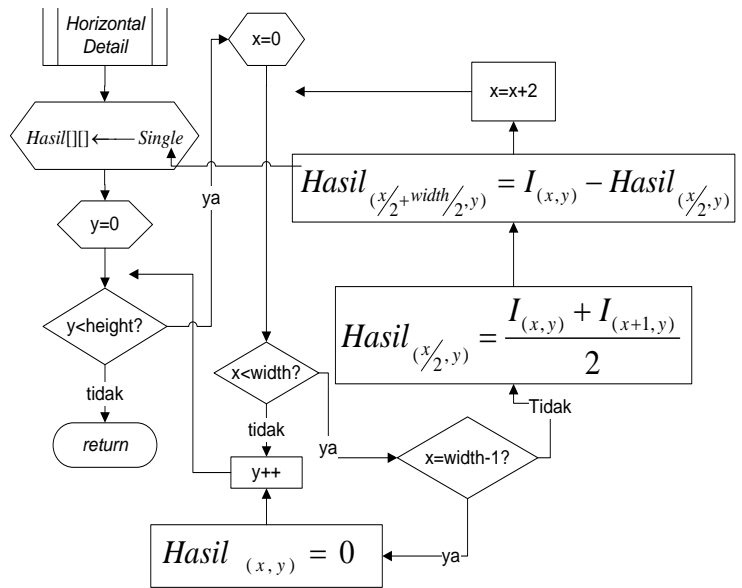
Haar Wavelet Transform

Pada tahapan proses *Haar Wavelet Transform* ini dilakukan 2 hal, yaitu *Haar* secara *horizontal (Horizontal detail)* dan *Haar* secara *vertical (Vertical detail)*, lihat gambar 4.



Gambar 4. Diagram Alir Proses *Haar Wavelet Horizontal Detail*

Pada proses *Horizontal detail* yang ditunjukkan pada gambar 5, setiap *pixel* pada kolom ganjil dioperasikan dengan *pixel* berikutnya pada baris yang sama. Kedua *pixel* tersebut akan dirata-rata dan dicari selisih antara *pixel* pertama dan rata-rata dari *pixel* tersebut. Apabila gambar berukuran ganjil, maka *pixel* terakhir akan diberi nilai 0. Rata-rata dari kedua *pixel* tersebut diletakkan di sisi kiri gambar, dan selisihnya diletakkan di sisi kanan gambar.



Gambar 5. Diagram Alir Proses *Horizontal Detail*

Vertical Detail

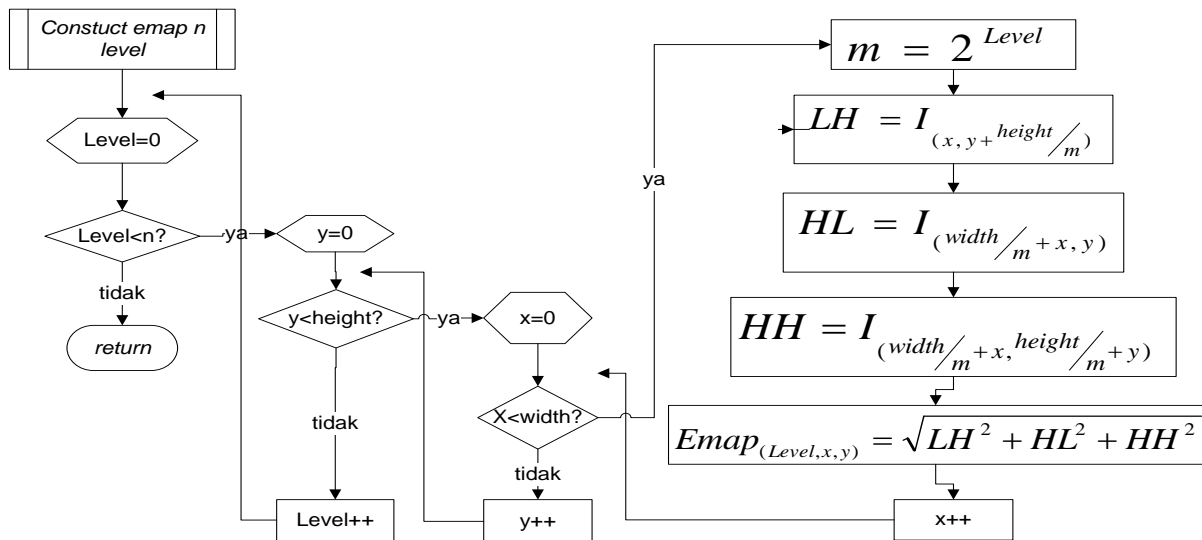
Pada proses *Vertical detail*, setiap *pixel* pada baris ganjil dioperasikan dengan *pixel* berikutnya pada kolom yang sama. Kedua *pixel* tersebut akan dirata-rata dan dicari selisih antara *pixel* pertama dan rata-rata dari *pixel* tersebut. Apabila gambar berukuran ganjil, maka *pixel* terakhir akan diberi nilai 0. Rata-rata dari kedua *pixel* tersebut diletakkan di sisi atas gambar, dan selisihnya diletakkan di sisi bawah gambar.

Construct Emap

Pada proses tahapan *Construct Emap*, *image* yang telah diubah dengan *Haar Wavelet* akan dihitung akar dari kuadran-kuadran LH, HL dan HH. LH berada pada sisi kiri bawah pada setiap *level* dalam *Haar Wavelet*, sedangkan HL berada pada sisi kanan atas dan HH berada pada sisi kanan bawah. LH, HL dan HH pada masing-masing *level* memiliki ukuran yang berbeda, sehingga akan menghasilkan *Emap* yang memiliki ukuran yang berbeda pula. Proses ini menggunakan persamaan 1. Diagram untuk pembuatan *Emap* ini dapat dilihat pada Gambar 6.

Blur analysis

Kemudian pada tahapan terakhir ini, dari *Emap* yang telah ada, tingkat blur akan dihitung berdasarkan *rules* yang ada (Tong, 2004).



Gambar 6. Diagram Alir Proses *Construct Emap*

4. HASIL PENGUJIAN SISTEM

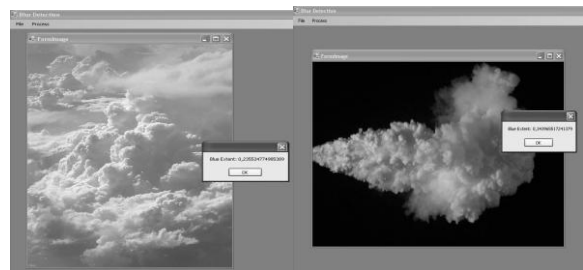
Beberapa pengujian dilakukan terhadap sebuah foto yang dikaburkan dengan menggunakan photoshop, metode *Gaussian Blur* dengan *radius* yang berbeda-beda. Hasil Pengujian tersebut dapat dilihat pada Tabel 5.

Tabel 5 Hasil pengujian terhadap gambar yang dikaburkan

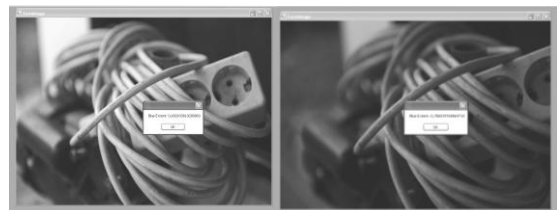
<i>radius</i>	Tingkat keaburan
1.0	0,16540
1.2	0,22377
1.4	0,25336
1.6	0,28732
1.8	0,33575
2.0	0,35936
2.2	0,40093
2.4	0,50788
2.6	0,55600
2.8	0,61314
3.0	0,65517

Proses pengujian selanjutnya dilakukan terhadap beberapa foto yang tidak blur, namun nampak seperti blur, yaitu gambar awan dan asap seperti pada Gambar 7. Dalam hal ini, foto dikenali sebagai foto yang blur dengan tingkat keaburan diatas 0.2 (foto awan) dan diatas 0.3 (foto asap). Pengujian juga dilakukan terhadap foto dari obyek yang sama namun berbeda tingkat intensitas, seperti nampak pada Gambar 8. hasil pengujian, foto

dengan intensitas lebih rendah akan memiliki tingkat keaburan yang lebih tinggi.

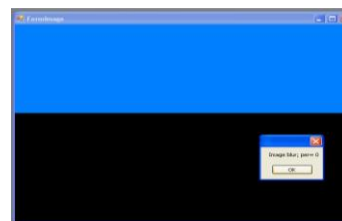


Gambar 7. Foto awan (kiri) dan asap (kanan)



Gambar 8. Foto dari obyek yang sama namun memiliki tingkat intensitas yang berbeda. Foto disebelah kiri memiliki intensitas lebih besar

Terakhir, pengujian dilakukan terhadap gambar dengan perubahan warna yang solid, seperti pada gambar 9. Pada gambar yang seperti ini, gambar dinyatakan blur.



Gambar 9. gambar dengan warna yang solid.

5. KESIMPULAN

Pendeteksian blur dilakukan dengan memperhitungkan jumlah area yang tidak kabur dari suatu gambar, sehingga tidak cocok untuk mendeteksi gambar fotografi dengan objek yang tidak kabur dan latar belakang yang kabur (gambar mikro). Metode ini mengenali tingkat blur berdasarkan garis yang ada pada foto. Jika sebagian kecil bagian dari foto terpisah oleh garis dengan sebagian kecil disekeliling bagian pertama, maka dianggap foto tidak blur.

Hal ini terlihat dari pengujian pada foto-foto yang tidak memiliki garis yang jelas, seperti awan dan asap, akan dideteksi dengan tingkat blur yang cukup tinggi. Hal ini juga terlihat pada pengujian gambar dengan warna yang solid. Karena pada warna yang sama tidak dikenali adanya garis, maka gambar dinyatakan blur.

Pengujian terhadap foto yang sengaja dibuat blur, menunjukkan bahwa perangkat lunak yang kami kembangkan ini sudah mampu mengenali tingkat blur sebagaimana yang diharapkan.

DAFTAR PUSTAKA

- [1] Sumbera, J. (2001). Wavelet Transform using Haar Wavelets. Retrieved August 3, 2011, diunduh dari http://aja.jikos.cz/vyplody/wavelet/Jiri_Sumbera_Wavelet_Transform_using_Haar_Wavelets.pdf
- [2] Li, Z., Drew, M. S. (2004). *Fundamentals of multimedia*.
- [3] Zulkaryanto, E. (2010). *Hough transform*. Retrieved September 10, 2011, from <http://zulkaryanto.files.wordpress.com/2010/01/hough-transform.pdf>
- [4] Tong, H., Li, M., Zhang H. & Zhang C. (2004). *Blur detection for digital images using wavelet transform*. Microsoft Research Asia. Retrieved August 3, 2011, from www.cs.cmu.edu/~htong/pdf/ICME04_tong.pdf
- [5] Weijer, J. V. D, Schmid, C. (2006). *Blur robust and color constant image description*. Retrieved August 5, 2011, from <http://lear.inrialpes.fr/people/vandeweyer/papers/icip06.pdf>