Studies in Gradient-Based Registration

Yacov Hel-Or

Internal Draft

1 Introduction

Assume we have two images I_1 and I_2 of the same scene taken from two different points of view. Image registration means warping I_1 towards I_2 in some fashion, so that the two images will be superimposed. Gradient-based registration relates to those techniques that find the warping parameters from the grayscale similarities between the two images [11]. Alternative approaches find the warping parameters using local features (such as, points, lines, etc.) extracted from the images (e.g. [10]).
The main advantage of the gradient-based techniques is that a correspondence between local features in the images is not required (finding the correspondence can be exponential in the number of features). However, the performance of gradient-based algorithms is adequate only in the cases where difference in the viewing position is relatively small and the illumination conditions in the two images are similar .

2 The Motion Model

Let $I_1(x, y)$ and $I_2(x, y)$ be two images of the same scene taken from different points of view. If the illumination condition is about the same in the two images and the viewing positions are not distant from each other we can assume the following:

$$I_1(x+u, y+v) = I_2(x, y) \quad . \tag{1}$$

i.e. the two images are shift invariant and the luminance of a physical point located at (x, y) in image I_2 is the same as the luminance of the same point in image I_1, which appears at the location $(x+u, y+v)$ (we ignore border problems). The vector field (u, v) defines the retinotopic (or image) motions between the two images, and it is a function of the location: $(u, v) = (u(x, y), v(x, y))$.

In general, one could represents an image motion in two ways: The first is to sample the functions (u, v) and represent its values over a discrete domain (i, j) where $(x, y) = (i\Delta x, j\Delta y)$. In most cases, $(\Delta x, \Delta y)$ are the width and height of the image pixel. Therefore, each pixel

(i, j) is associated with two values: $u_{i,j}$ and $v_{i,j}$ describing the pixel motion in the vertical and in the horizontal directions. With this description there is no necessary relationship between motions of neighboring pixels; $u_{i,j}$ and $v_{i,j}$ represent local motion of a particular pixel. This representation is called a *local parametric* motion model [4]. Although, this representation is simple, it suffers from two main drawbacks. First, it does not exploit the fact that the image motion is projected from a realistic scene and has some global characteristics. Second, the motion description is not compact. It requires estimation of $2n$ unknowns (for n pixels). Since each pixel generates one equation (Equation 1) we have only n equations obtaining an under-determined system of equations. This is a version of the well known "aperture problem" [9] in which we have infinite solutions for any input. In order to make the system solvable we must constrain $(u_{i,j}, v_{i,j})$ according to some a priori knowledge. One possible constraint is related to the assumption that the depth of a regular scene does not change rapidly relative to the image plane, so that the retinotopic motion is a "smooth" function. This is known as the *regularization* constraint [14, 15] in which the "smoothest" solution is chosen from amongst the infinite possible solutions.

An alternative representation is to formulate the motion (u, v) as two functions $u(x, y, \mathbf{p})$ and $v(x, y, \mathbf{p})$ where (x, y) is the a continuous domain and $\mathbf{p}$ is a parameter vector. The parameter vector $\mathbf{p}$ is unknown and needs to be estimated. Since the parametric functions u and v describe the entire image motion, this kind of motion description is called a *global parametric* motion model [4]. Using the global parametric motion model requires assumptions on the scene structure. Possible assumptions that are widely used are:

- The scene is rigid.
- Known parametric structure [12]. For example, a planar scene can be described by the equation $Z = aX + bY + c$ where a, b and c are unknown parameters, and (X, Y, Z) are the world coordinates. Other parametric structures can be: piecewise linear, k degree polynomial, B-spline, etc.
- Known structure, i.e. the $3D$ coordinates of each scene point is known relative to some arbitrary chosen basis or the depth of each pixel relative to the image plane is known for one of the images.

When dealing with scenes that are easy to parametrize it is advantageous to use the global parametric motion model. The following will be concentrated on this kind of motion model.

3 Global Parametric Motion Model

As stated in the first chapter, a basic assumption about gradient-based registration is that the two images are taken from relatively close points of view. This assumption results from the grayscale invariant assumption stated in Equation 1. The small displacement assumption enables us to view the two image registration problem as a motion registration problem where

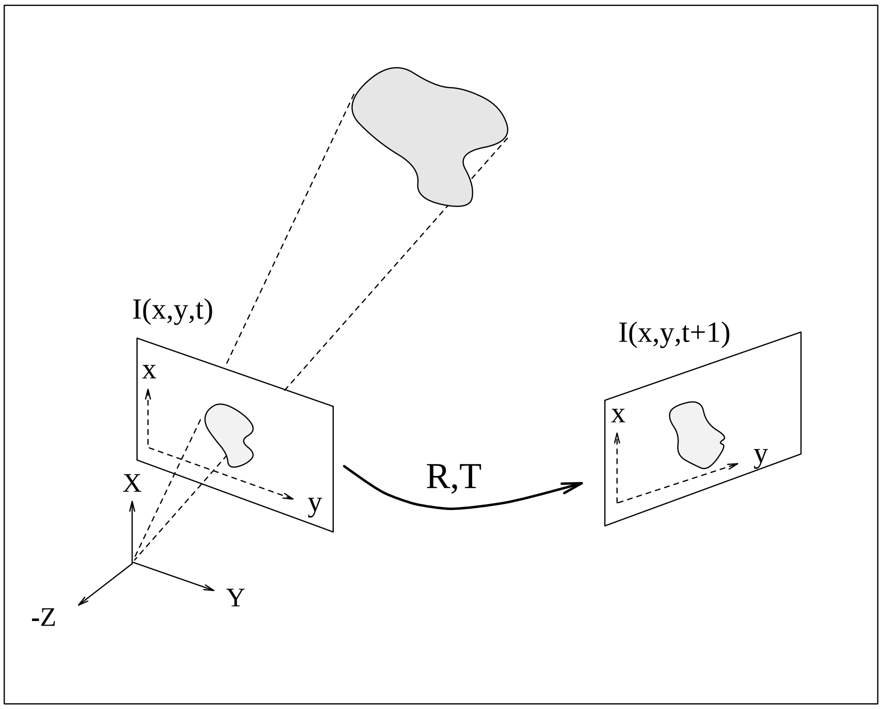

Figure 1: Two images taken from a moving camera.

a sequence of images $I(x, y, t)$ are collected from a moving camera and registration is required for every pair of successive images $I(x, y, t)$ and $I(x, y, t + \Delta t)$. This terminology will be used in the following sections.

3.1 The Rigidity Constraint

Assume a pin-hole camera is translated by a translational velocity $\mathbf{T}$ and angular velocity $\mathbf{R}$ (see Figure 1). If we assume that the scene is rigid, the $3D$ motion of a 3D point $\mathbf{P}(t) = (X(t), Y(t), Z(t))^T$ can be described in the camera coordinate frame by:

$$\frac{\partial \mathbf{P}}{\partial t} = \mathbf{P}(t + \Delta t) - \mathbf{P}(t) = -\mathbf{T} - \mathbf{R} \times \mathbf{P}(t)$$

Expanding this into components yields:

$$\begin{aligned} \frac{\partial X}{\partial t} &= -T_z - R_y Z(t) + R_y Y(t) \\ \frac{\partial Y}{\partial t} &= -T_y - R_z X(t) + R_x Z(t) \\ \frac{\partial Z}{\partial t} &= -T_z - R_x Y(t) + R_y X(t) \end{aligned} \tag{2}$$

Projecting $\mathbf{P}$ onto the image plane (x, y) gives [9]:

$$x = \frac{fX}{Z} \quad ; \quad y = \frac{fY}{Z}$$

The derivative of (x, y) with respect to time is then:

$$\begin{aligned} u &= x_t = \frac{f}{Z^2}(X_t Z - X Z_t) \\ v &= y_t = \frac{f}{Z^2}(Y_t Z - Y Z_t) \end{aligned}$$

where s_t for any s represents the discrete derivative of s with respect to time: $\frac{s(t+\Delta t)-s(t)}{\Delta t}$. Substituting the derivatives from equation 2 into the above equations yields:

$$\boxed{(u, v)^T = \frac{1}{Z(x, y, t)} AT + BR \qquad (3)}$$

where

$$A = \begin{bmatrix} -f & 0 & x \\ 0 & -f & y \end{bmatrix}$$

and

$$B = \begin{bmatrix} xy/f & -(f + x^2/f) & y \\ (f + y^2/f) & -xy/f & -x \end{bmatrix}$$

The matrices A and B are known but $\mathbf{T}$, $\mathbf{R}$ and $Z(x, y)$ are unknown and should be estimated. If it is difficult to represent the scene by a parametric function, one could represents $Z(x, y)$ over the discrete domain (i, j). In this case, the number of unknowns to be estimated is $6 + n$ (n is the number of pixels); n depth parameters $Z(i, j)$, and 6 motion parameters determining $\mathbf{T}$ and $\mathbf{R}$. Note that the rigidity constraint reduces the number of unknowns from $2n$, as in the local parametric model, to $6 + n$ unknowns.

Note:

- The depth $Z(x, y)$ and the translation $\mathbf{T}$ can be estimated up to a scale factor (since $\mathbf{T}$ is divided by Z in Equation 3). Therefore, in fact we can recover only $5+n$ unknowns.
- The contribution of the camera rotation to (u, v) is independent of the depth $Z(x, y)$, while the contribution of the camera translation does depend on the depth.

3.2 Known Depth

In the case where the depth $Z(i, j)$ is known for every pixel location (i, j):

$$(u, v)^T_{i,j} = A_{i,j} T + BR$$

where $A_{i,j} = A/Z(i, j)$ is a known matrix. In this case there are only 6 unknowns to be found - the motion parameters.

3.3 Parametric Structure

A view of an airport from a landing airplane can be regarded as a quasi-planar scene. The following discussion will concentrate on this kind of structure.

3.3.1 A Plane Under Perspective Projection

A planar scene can be easily represented by a parametric function. All points of this plane satisfy the parametric equation:

$$Z = a + bX + cY$$

where (X, Y, Z) are some arbitrary chosen world coordinates. Dividing the above equation by aZ we get:

$$\frac{1}{Z} = \frac{1}{a} - \frac{bX}{aZ} - \frac{cY}{aZ}$$

which can be rewritten as:

$$\frac{1}{Z} = \alpha + \beta x + \gamma y$$

and therefore (from Equation 3):

$$(u, v)^T = (\alpha + \beta x + \gamma y)A\mathbf{T} + B\mathbf{R}$$

In this motion model we have 9 unknowns to be found. Since (α, β, γ) and $\mathbf{T}$ are coupled, it is impossible to find the motion parameters from the displacement of one plane, and at least one additional plane is required.
However, for the registration problem, it is enough to find the 8 uncoupled parameters which can be written as (see [2, 4] for more details):

$$\begin{aligned} u &= a_1 + a_2 x + a_3 y + (a_4 x + a_5 y)x \\ v &= a_6 + a_7 x + a_8 y + (a_4 y + a_5 y)y \end{aligned}$$

This motion model has been called the "Pseudo Projective Model" [13].

3.3.2 A Plane Under Orthographic Projection

If an object covers a small region in the field of view, and is far away from the camera (relative to its diameter), its projection can be approximated by an orthographic projection [9]:

$$x = sX \quad ; \quad y = sY$$

where $s = f/Z_0$ is a constant and Z_0 is the average depth of the object. Also, in this case:

$$\begin{aligned} u &= \frac{\partial x}{\partial t} = s\frac{\partial X}{\partial t} = s(-T_z - R_y Z + R_y Y) \\ v &= \frac{\partial y}{\partial t} = s\frac{\partial Y}{\partial t} = s(-T_y - R_z X + R_x Z) \end{aligned}$$

When dealing with a plane we get:

$$\begin{aligned} u &= s(-T_z - R_y(a + bX + cY) + R_y Y) \\ v &= s(-T_y - R_z X + R_x(a + bX + cY)) \end{aligned}$$

which can be rewritten as an affine transformation in the image plane [4]:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_5 \\ a_6 \end{pmatrix}$$

In this case there are only 6 parameters to be found.

4 Multiresolution Estimation

In any chosen motion model (global parametric) we have:

$$I(x + u(x, y, \mathbf{p}), y + v(x, y, \mathbf{p}), t) = I(x, y, t + \Delta t) \quad .$$

We aim to find the parameter vector $\mathbf{p}$ which minimizes

$$E = \sum_{x,y} J(I(x + u(x, y, \mathbf{p}), y + v(x, y, \mathbf{p}), t) - I(x, y, t + \Delta t))$$

where J is some penalty function. Since $I(x + u, y + v)$ depends non-linearly on $\mathbf{p}$ we have a non-linear minimization problem at hand. One approach to this non-linear minimization is the multiresolution technique [4, 3]. The following is a description of the multiresolution minimization technique, its advantages and its limitations.

For simplicity, assume the motion model we choose consists of one parameter - a translation in the x direction. In the example shown in Figure 2 two images were displaced from each other by 100 pixels in the x direction. The energy function (square penalty) as a function of p is shown in Figure 3:

$$E(p) = (I(x + p, y, 1) - I(x, y, 2))^2 \tag{4}$$

It can be seen from this energy function that there is a single deep global minimum with a narrow basin of attraction and many other local minima. An exhaustive search for the minima is possible but it is time consuming (exponential with the number of parameters). A gradient based minimization can reduce the expensive exhaustive search. We apply the minimization using several starting points in the parameter domain. The range in parameter space where the minimization will converge to the global minimum is the basin of attraction (B.O.A.) of the solution (i.e. the global minimum in this range is the only local minimum). In order to guarantee convergence, intervals between starting points should be less than or

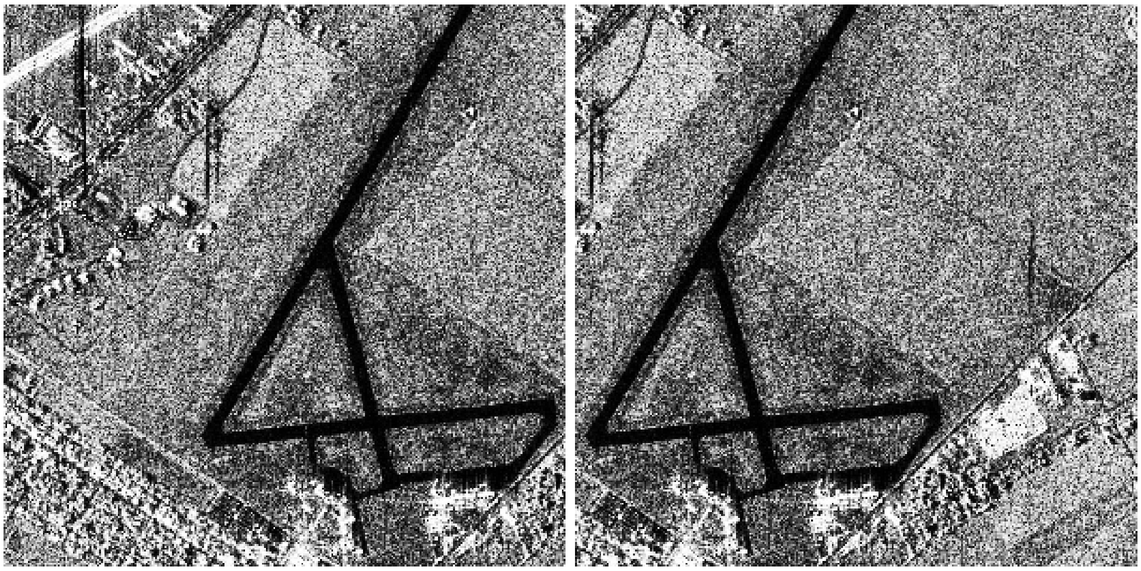

Figure 2: Two images translated from each other by 100 pixels.

equal to the width of B.O.A. Therefore, run-time of a gradient based minimization depends on the width of the basin of attraction. As the B.O.A is wider the performance improves.

Expansion of the basin of attraction (B.O.A) can be done using multiscale techniques (pyramids): Given the original image $I^0(x,y)$, we generate several lower resolution images $I^k(x,y)$ where the image at level k is generated from the image at level $k-1$. Two types of pyramids are commonly used: the *Gaussian* pyramid and the *Laplacian* pyramid [6, 5]. In the Gaussian pyramid we generate the next level by convolving the current level with a Gaussian kernel followed by subsampling:

$$I^k(x,y) = S \cdot G * I^{k-1}(x,y)$$

where

$$G = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

The convolution with G is a low pass filter and S is a subsampling (in general, having a sampling rate of 1:2). This is equivalent to a low pass filtering. The standard deviation σ of the Gaussian depends on the Nyquist frequency. If the subsampling rate is 1:2 then we want to multiply the frequency domain with a Gaussian which attenuates frequencies above $n\Delta u/2$ (Δu is the interval between two neighboring samples in the frequency domain and $n\Delta u$ is the frequency range of the image). The Gaussian with $\sigma_u \approx n\Delta u/4$ will be appropriate. This multiplication is equivalent to a convolution in the spatial domain with a Gaussian of $\sigma_x = \frac{4}{n\Delta u} = \frac{4n\Delta x}{n} = 4\Delta x$ (Δx is the interval between two neighboring samplaes in the spatial domain).

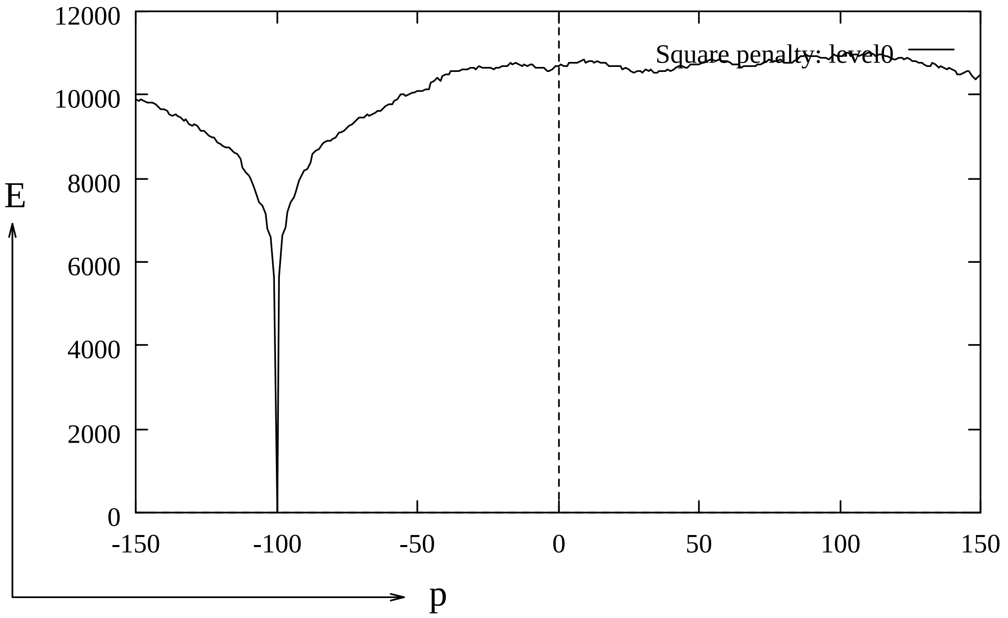

Figure 3: Energy function relative to the translation parameter p.

In the Laplacian pyramid we generate the next level by subtracting the current level of the Gaussian pyramid from the (expanded) next level of the Gaussian pyramid:

$$L^k(x,y) = expand\{I^{k+1}(x,y)\} - I^k(x,y)$$

This is equivalent to a band pass filtering. The advantages and disadvantages of using each one of these pyramids will be discussed later.

An example of the Gaussian pyramid with 4 levels, generated from the image of Figure 2 can be seen in Figure 4.

In order to find the translation parameter between two images, a pyramid is generated from each of the images and an energy function $E_k(p)$ is calculated for every pyramid level k:

$$E_k(p) = (I^k(x+p,y,t) - I^k(x,y,t+\Delta t))^2$$

This process was applied to the images of Figure 2. The energy functions $E_k(p)$ obtained for 5 pyramid levels are shown in Figure 5.
It can be seen from these energy functions that as the level in the pyramid increases (the resolution decreases)

- the basin of attraction expands.
- the sharpness of the energy function at the global minimum p_{min} decreases and so does the precision in localizing the global minimum.

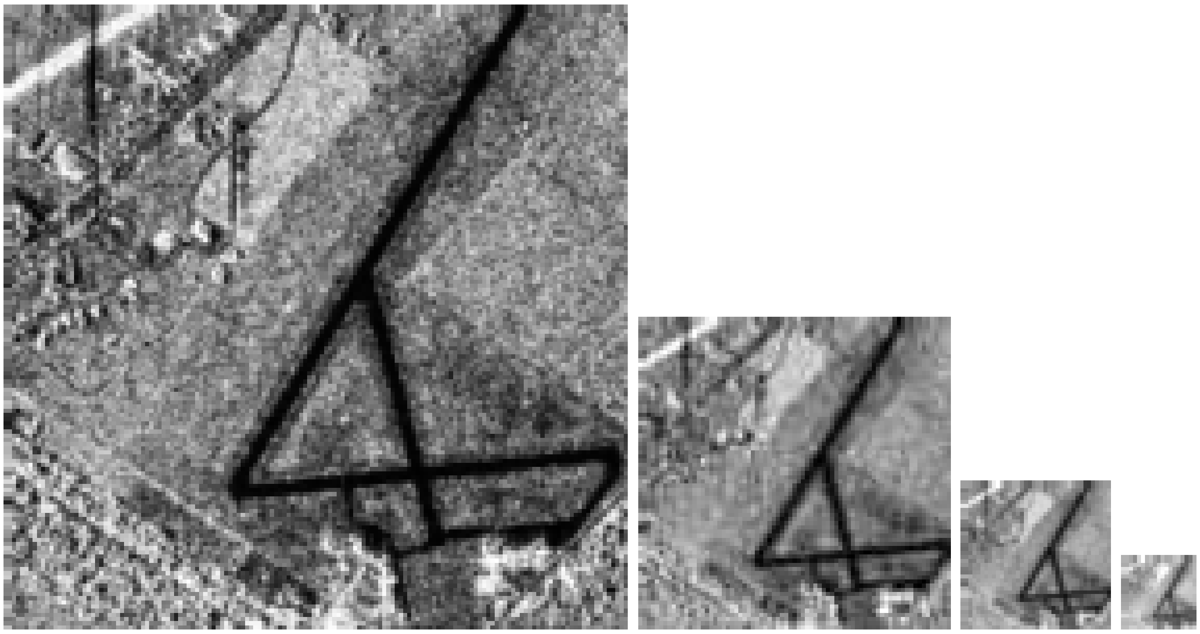

Figure 4: Gaussian Pyramid with 4 levels.

Taking advantage of this behavior, the motion estimation between two successive images I_1 and I_2 can proceed as follows:

1. Generate two pyramids $\{I_1^k\}$ and $\{I_2^k\}$, $k = 0 \cdots N$ where N is the highest level.
2. Find an initial solution for the motion parameters by finding the global minimum of $E_N(p)$. This is done by sparsely sampling the parameter space and converging to a local minimum in $E_N(p)$ from each sample. Among the local minima choose the parameter of the global minimum as the initial solution.
3. Transform the parameter associated with the global minima of the current level to a parameter associated with the lower level.
4. At the lower level- search for a local minimum from the given initial guess.
5. Go to step 3 until level 0 is reached.

The above algorithm can be generalized to any parameter space $\mathbf{p}$. Three examples of finding the registration parameters using the above algorithm follow. In Figure 6 the pictures differ by a shift in the x direction. In Figure 7 the pictures differ by a shift in the x and in the y directions. In the third example, Figure 8, the pictures are related to each other by an affine transformation. In all these cases the algorithm found the correct transformations.

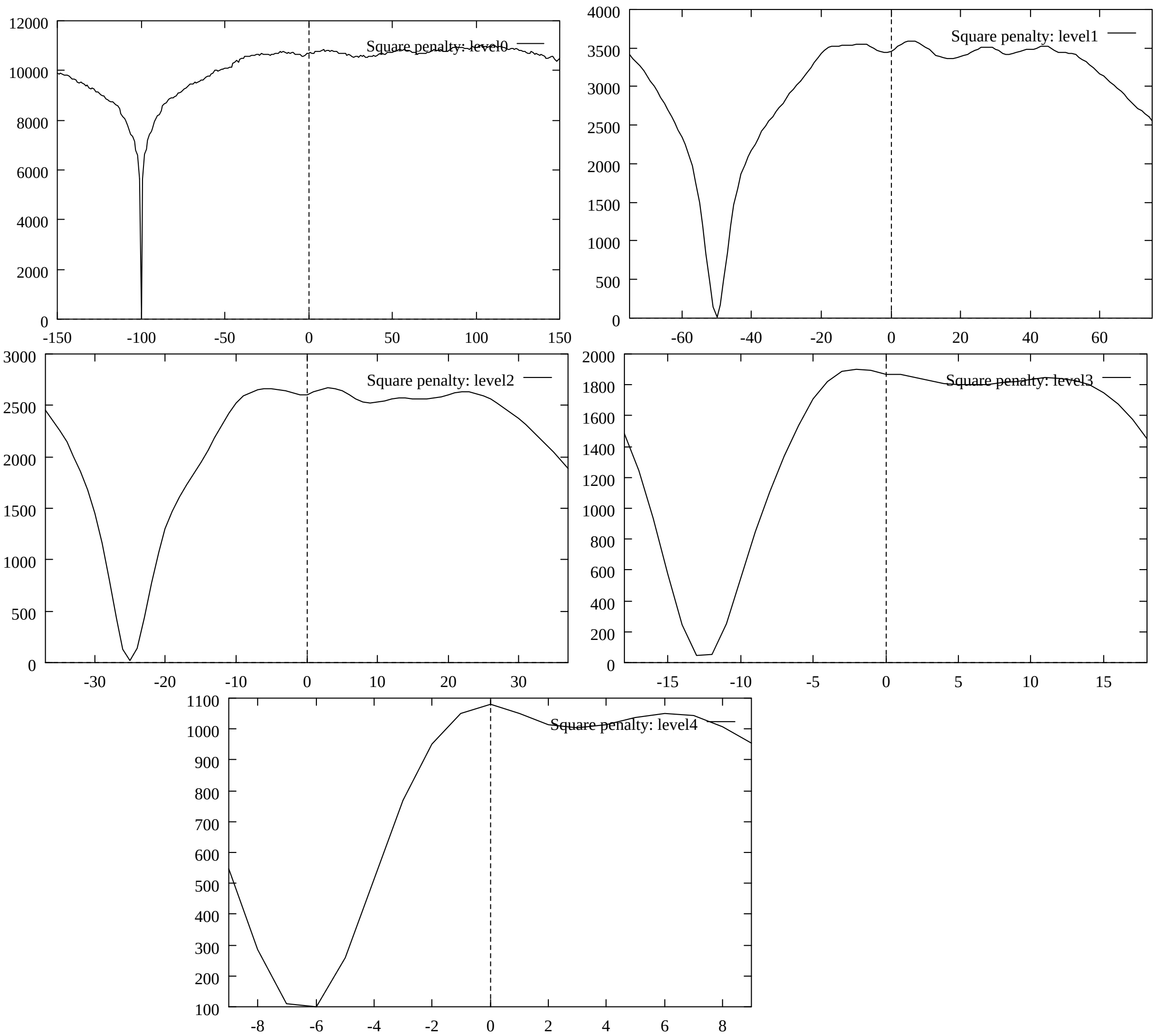

Figure 5: Five energy functions for five pyramid levels.

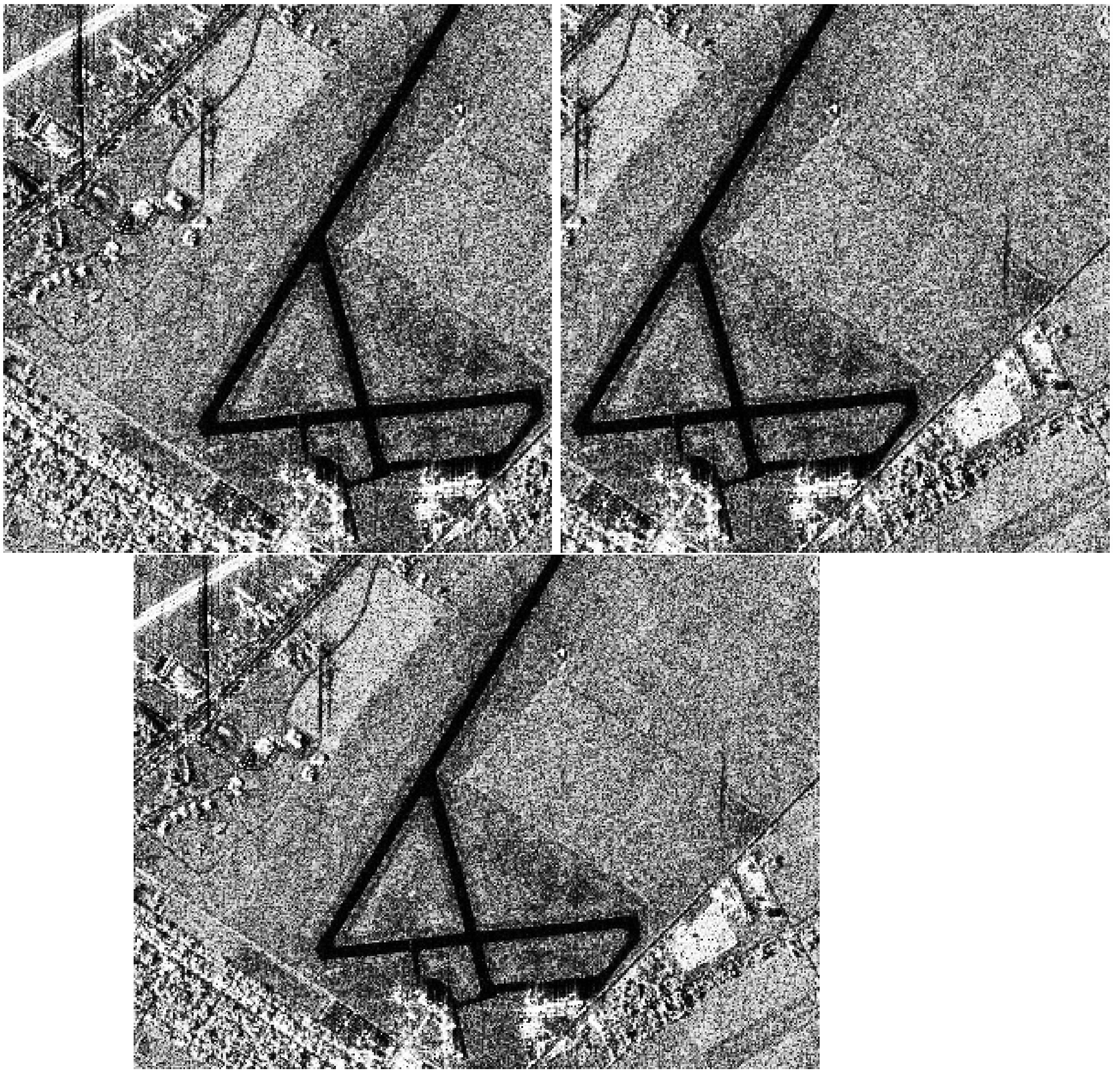

Figure 6: Registration for one dimensional translation.

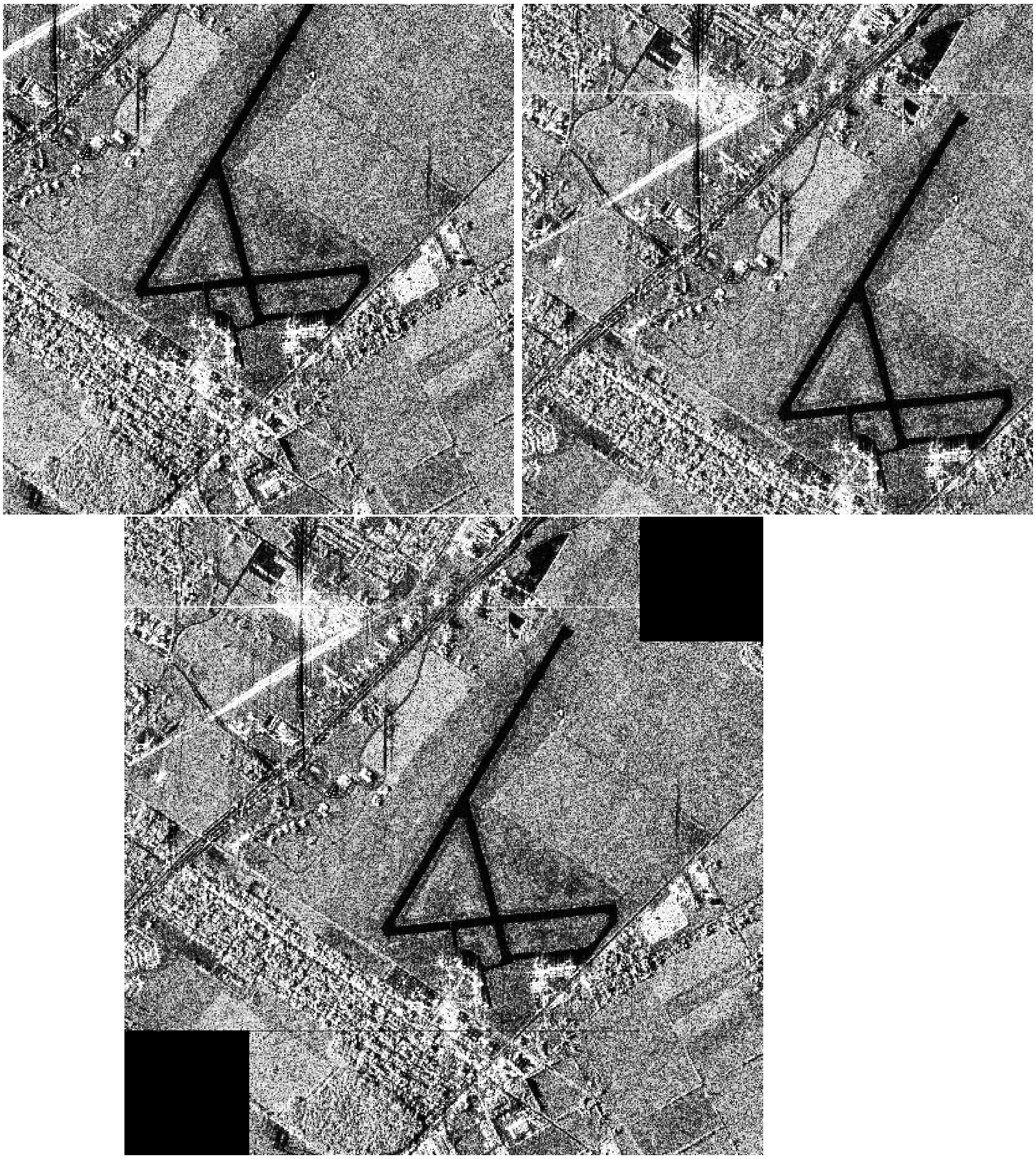

Figure 7: Registration for two dimensional translation.

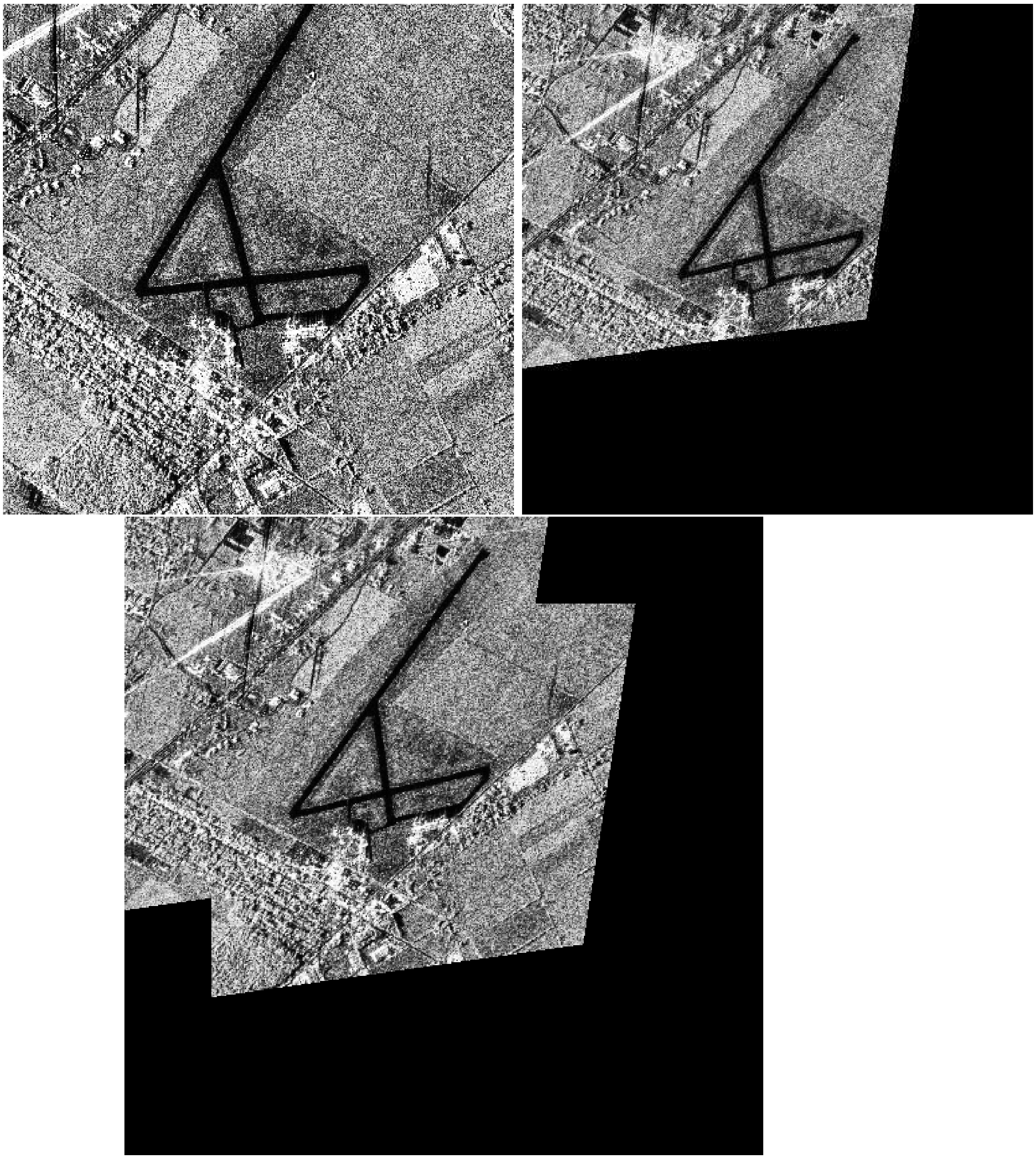

Figure 8: Registration for affine transformation.

5 Performance and Convergence Guarantee

In the previous section an important advantage of the multiresolution minimization technique was described: The ability to reduce computational time in searching for the global minimum in the parameter space. The question which will be clarified in this section is whether this is the only contribution of the multiresolution scheme. In other words, if we have unlimited computational abilities - do we need the multiresolution technique? In this section, we would also like to clarify whether the convergence of the proposed algorithm is guaranteed and in what circumstances. In other words, how dense do we have to sample the parameter space while searching for an initial guess in the algorithm (step no. 2 in the algorithm) in order to guarantee convergence to the true solution.

In order to investigate these questions, consider the simple example of one dimensional function translated at a constant velocity:

$$I(x,t) = I_0(x - pt)$$

Let us consider x and t to be continuous variables. In the spatio-temporal space this motion appears as slanted strips with a slope inversely proportional to the velocity and equal to $1/p$ (Figure 9).

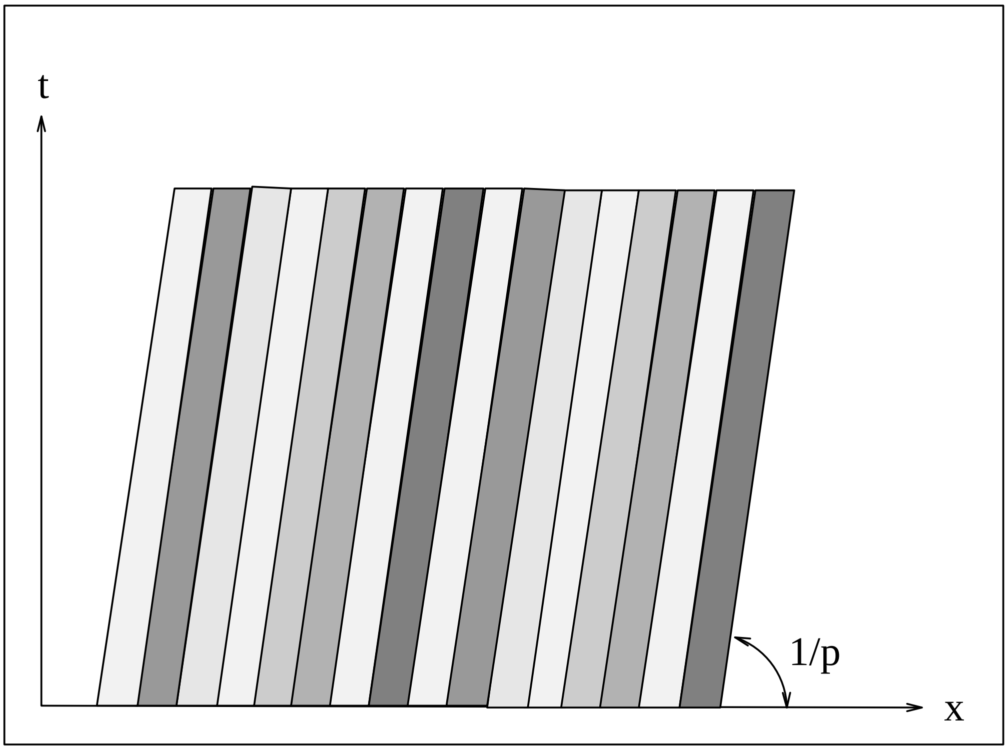

Figure 9: Spatio-temporal representation of a translated one dimensional image.

Let the spatio-temporal frequency domain of I(x,t) be $\hat{I}(u,w)$:

$$\begin{aligned}
\hat{I}(u,w) &= \int\int I_0(x-pt)\exp^{-2\pi i(ux+tw)}\,dxdt \\
&= \int\int I_0(x-pt)\exp^{-2\pi iux}\exp^{-2\pi itw}\,dtdx \\
&= \int \hat{I}_0(u)\exp^{-2\pi iupt}\exp^{-2\pi iwt}\,dt \\
&= \hat{I}_0(u)\int \exp^{-2\pi it(w+up)}\,dt \\
&= I_0(u)\delta(w+up)
\end{aligned}$$

This equation defines a one dimensional line $I_0(u)$ in the frequency domain passing through the origin and having a slope of $-p$, equal in its magnitude to the velocity value. This is demonstrated by the black line in Figure 10.

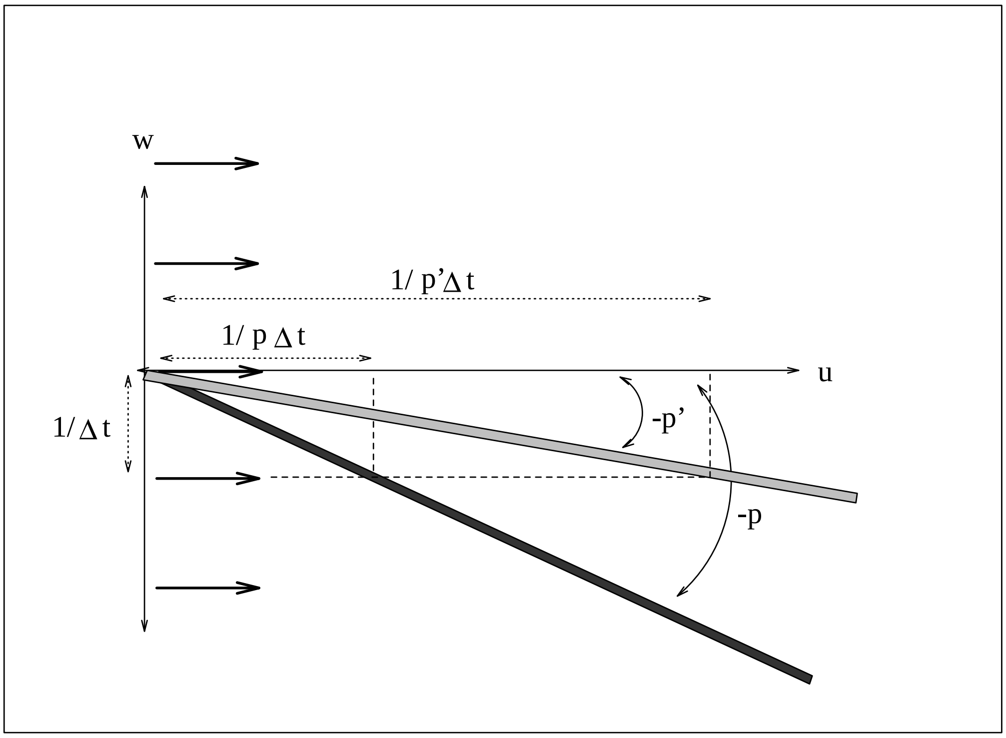

Figure 10: A moving image is represented by a slanted line in the frequency spatio-temporal domain, with a negative slope equal to the speed. The values along this line are equal to the static Fourier values $I_0(u)$.

Assume that we want to recover the speed p by minimizing the energy function:

$$E(v) = \int\int [I(x,t) - I(x+v\delta t, t+\delta t)]^2 dxdt \tag{5}$$

If the time interval δt is small enough, expanding $I(x+v\delta t, t+\delta t)$ around (x,t) and taking the first order approximation yields:

$$I(x+v\delta t, t+\delta t) = I(x,t) + v\delta t I_x + \delta t I_t \tag{6}$$

where

$$I_x = \frac{\partial I(x,t)}{\partial x} \quad ; \quad I_t = \frac{\partial I(x,t)}{\partial t}$$

Substituting Equation 6 into Equation 5 and defining $\delta t = 1$ yields:

$$E(v) = \int\int [vI_x + I_t]^2 dxdt = \int\int [v^2 I_x^2 + I_t^2 + 2vI_x I_t] dxdt$$

Expressing the above equation in the frequency domain gives:

$$\begin{aligned}
\int\int I_x^2 dxdt &= -4\pi^2 \int\int u^2 |\hat{I}(u,w)|^2 dudw \\
\int\int I_t^2 dxdt &= -4\pi^2 \int\int w^2 |\hat{I}(u,w)|^2 dudw \\
\int\int I_x I_t dxdt &= -4\pi^2 \int\int uw |\hat{I}(u,w)|^2 dudw
\end{aligned}$$

Therefore the above minimization can be expressed as a minimization in the frequency domain:

$$E(v) = \int\int [vu + w]^2 |\hat{I}(u,w)|^2 dudw$$

Since in our case $|\hat{I}(u,w)|^2 \neq 0$ only when $w = -pu$ (p is the speed of motion), we get:

$$E(v) = \int\int (v-p)^2 u^2 |\hat{I}(u,w)|^2 dudw \tag{7}$$

This quadratic function is plotted in Figure 11 and obtains its minimum at $v = p$.

W.l.o.g. assume that we sample the function $I(x,t)$ every Δt in order to recover the speed p from the image sequence. Sampling the spatio-temporal domain every Δt in the time space is equivalent to convolving the frequency domain with a comb function of spacing $1/\Delta t$. This replicates the frequency image along the w axis (see Figure 12). However, if the sampling rate is lower than a certain rate, more than one replica will intersect the integration domain (marked by the dashed line in Figure 12) and will deteriorate the energy function. Simple geometry calculations shows that in order to prevent this aliasing effect we have to eliminate all frequencies (in the spatial domain) above $1/(2p\Delta t) = 1/2D$, where D is the displacement between two consecutive images. In other words we have to ensure that the maximum frequency in the spatial domain u_{max} will be:

$$u_{max} \leq \frac{1}{2p\Delta t} = \frac{1}{2D}$$

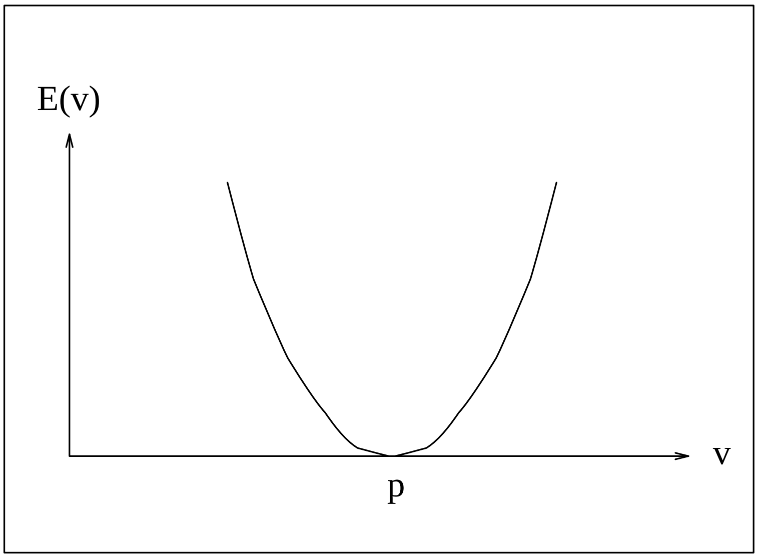

Figure 11: The energy function $E(v)$ as a function of v.

As p or Δt increases (i.e. the displacement between two images increases) the permitted maximum frequency u_{max} decreases.

Now we can understand the advantage of the multi-resolution techniques on top of the computational reduction. As we go to higher levels in the pyramid representation, the maximum spatial frequency decreases and therefore a larger displacement between images can be detected. Note, however, that in order to guarantee convergence we must have enough energy in the frequency band $[0, 1/(2p\Delta t)]$.

The above study can also give us an insight about the basin of attraction (B.O.A) of the energy function $E(p)$ (Equation 4) at each pyramid level. Since we expect to properly approximate the slope in the frequency domain when $u_{max} \leq 1/2D$ it is clear that for a given u_{max} the basin of attraction will be at least $2D$ i.e. the B.O.A will be greater or equal to $1/u_{max}$. This tells us that if no a priori information is given about the solution, we have to sample the parameter space at least every $1/u_{max}$ in order to guarantee a conversion to the true solution.

An example of predicting the B.O.A. in different frequency bands can be seen in Figures 13-27. In these examples, energy functions $E(p)$ were calculated between two equal images (i.e. the translation between the images is zero) after filtering them with band pass filters. The original image is shown in Figure 13. This image contains 256 pixels, therefore, each Δu is equal to $1/(256\Delta x)$. If $k\Delta u = u_{max}$ is the maximum frequency of this image, the B.O.A. is predicted to be greater or equal to:

$$\frac{1}{k\Delta u} = \frac{256\Delta x}{k}$$

which means $256/k$ pixels. The energy functions for 6 different frequency bands are shown

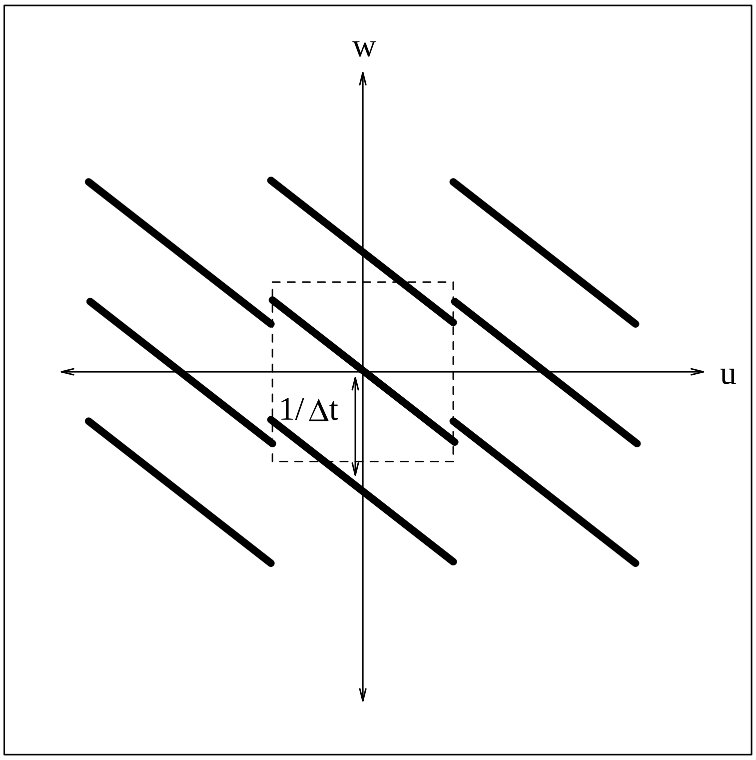

Figure 12: The frequency domain when sampling $I(x,t)$ every Δt.

in Figures 14-27. The predicted B.O.A. and the calculated B.O.A. for each case are summarized in the following table:

Figure no.	Frequency Band	Predicted B.O.A	calculated B.O.A
12	$[0\Delta u, 5\Delta u]$	$51\Delta x$	$54\Delta x$
14	$[5\Delta u, 10\Delta u]$	$25\Delta x$	$26\Delta x$
16	$[10\Delta u, 15\Delta u]$	$17\Delta x$	$16\Delta x$
18	$[15\Delta u, 20\Delta u]$	$13\Delta x$	$12\Delta x$
20	$[20\Delta u, 25\Delta u]$	$10\Delta x$	$10\Delta x$
22	$[25\Delta u, 30\Delta u]$	$9\Delta x$	$8\Delta x$
24	$[30\Delta u, 35\Delta u]$	$7\Delta x$	$8\Delta x$

Due to the digitization behavior in the energy function we can have an error in the calculated B.O.A. of up to 1 pixel. Taking this into account, it can be seen from the table that the real B.O.A. is wider or equal to the predicted B.O.A. as expected.

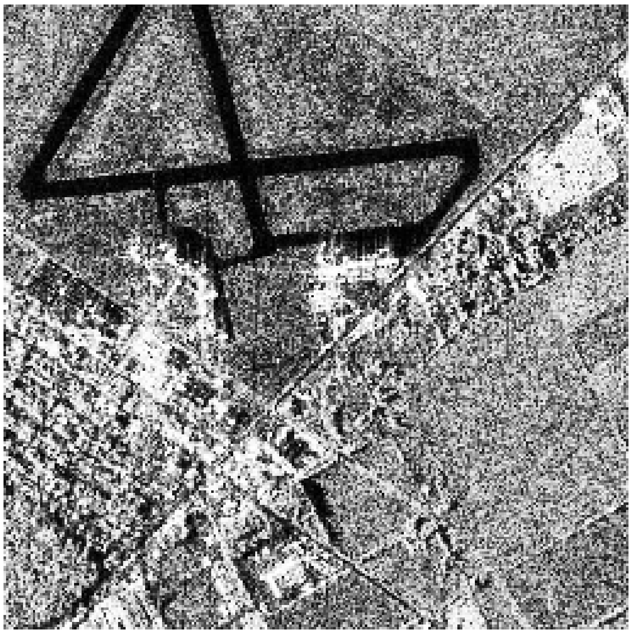

Figure 13: The original image before filtering by a band pass filter. This image contains 256x256 pixels.

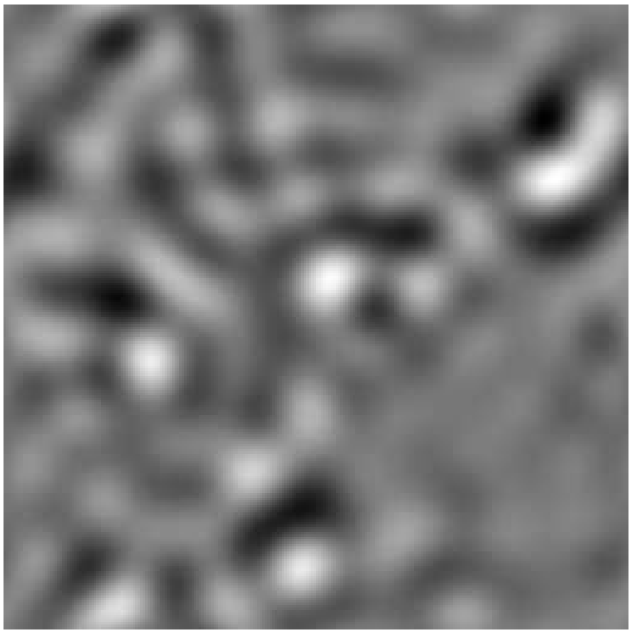

Figure 14: Band pass image with frequencies $0\Delta u \leq u \leq 5\Delta u$.

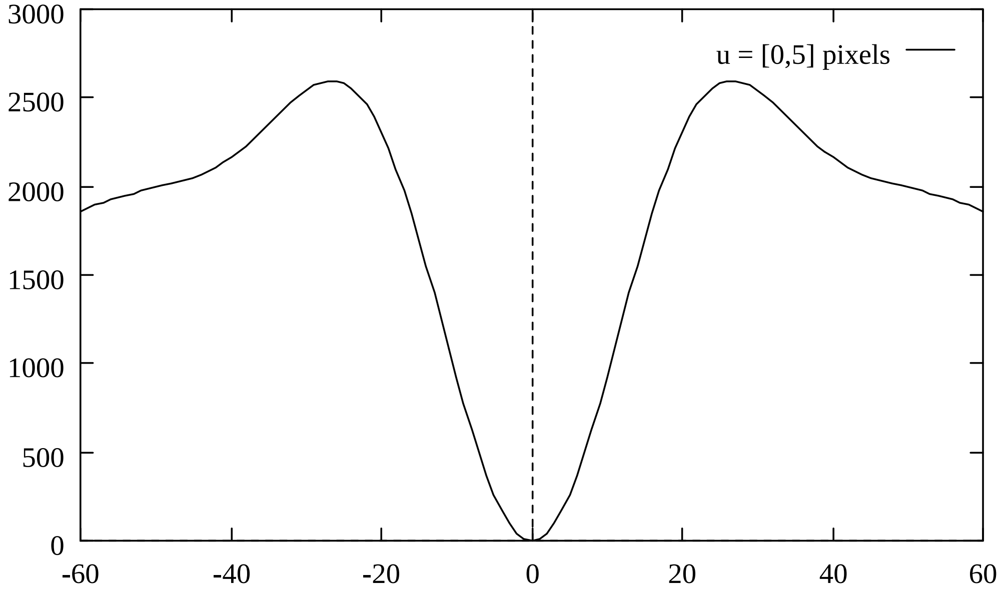

Figure 15: The energy function of the above image. The calculated B.O.A is 54.

Figure 16: Band pass image with frequencies $5\Delta u \leq u \leq 10\Delta u$.

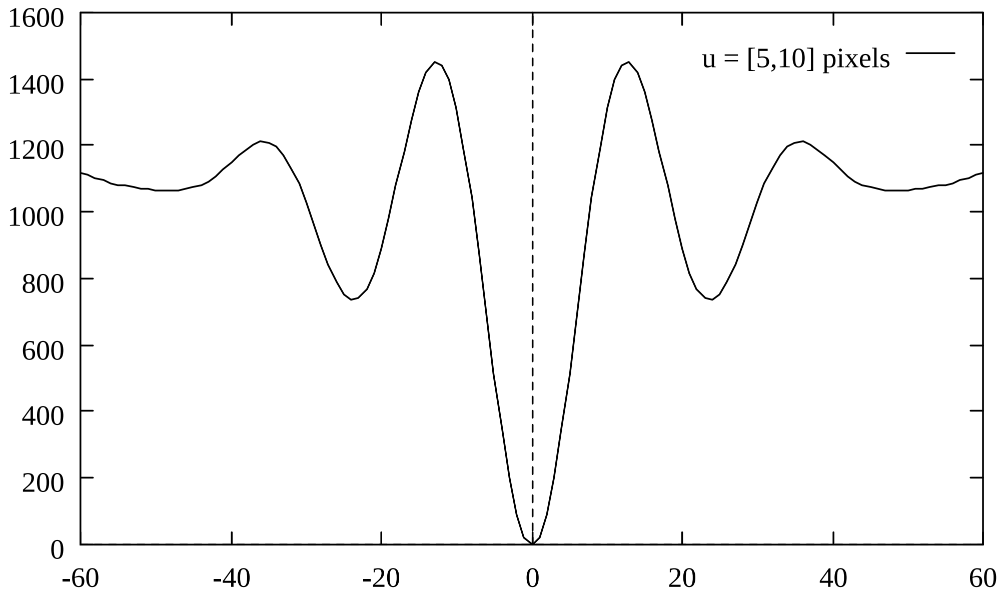

Figure 17: The energy function of the above image. The calculated B.O.A is 26 pixels.

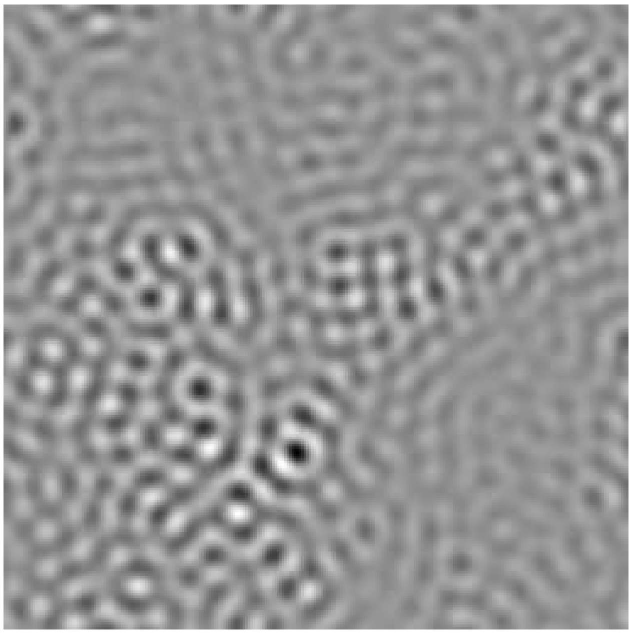

Figure 18: Band pass image with frequencies $10\Delta u \leq u \leq 15\Delta u$.

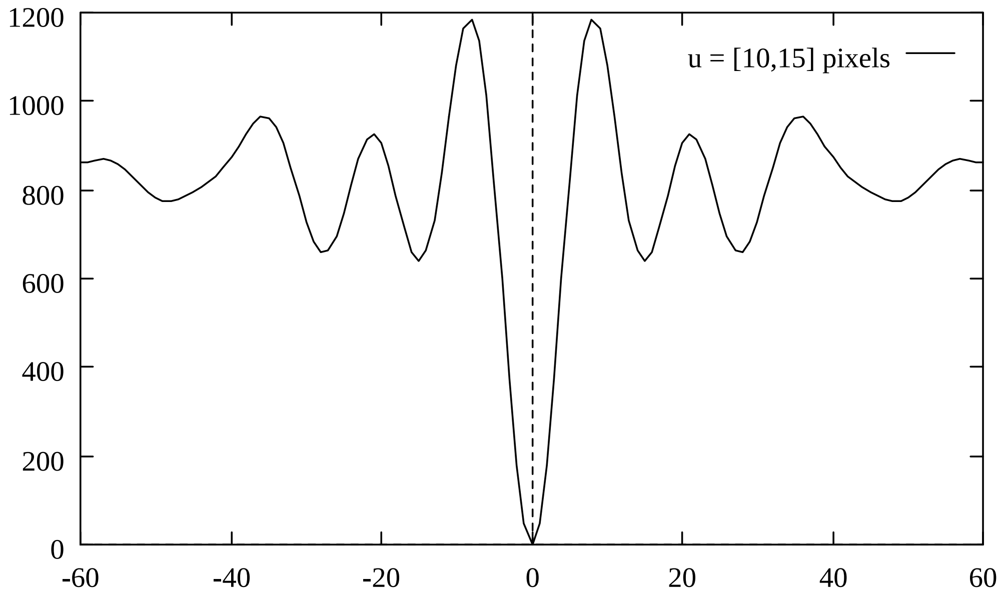

Figure 19: The energy function of the above image. The calculated B.O.A is 16 pixels.

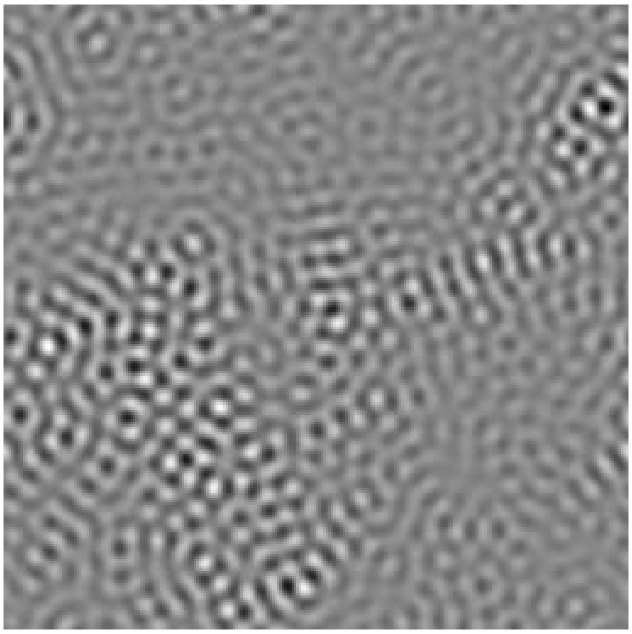

Figure 20: Band pass image with frequencies $15\Delta u \leq u \leq 20\Delta u$.

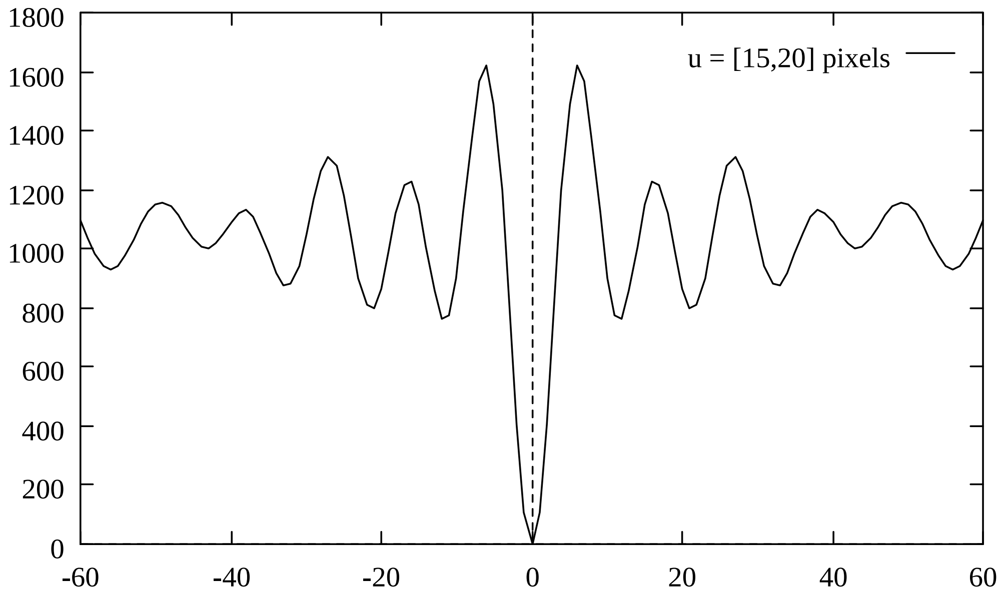

Figure 21: The energy function of the above image. The calculated B.O.A is 12 pixels.

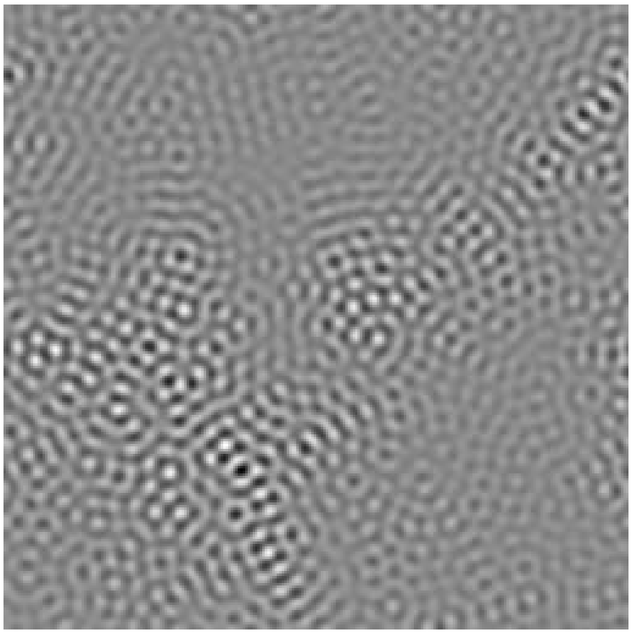

Figure 22: Band pass image with frequencies $20\Delta u \leq u \leq 25\Delta u$.

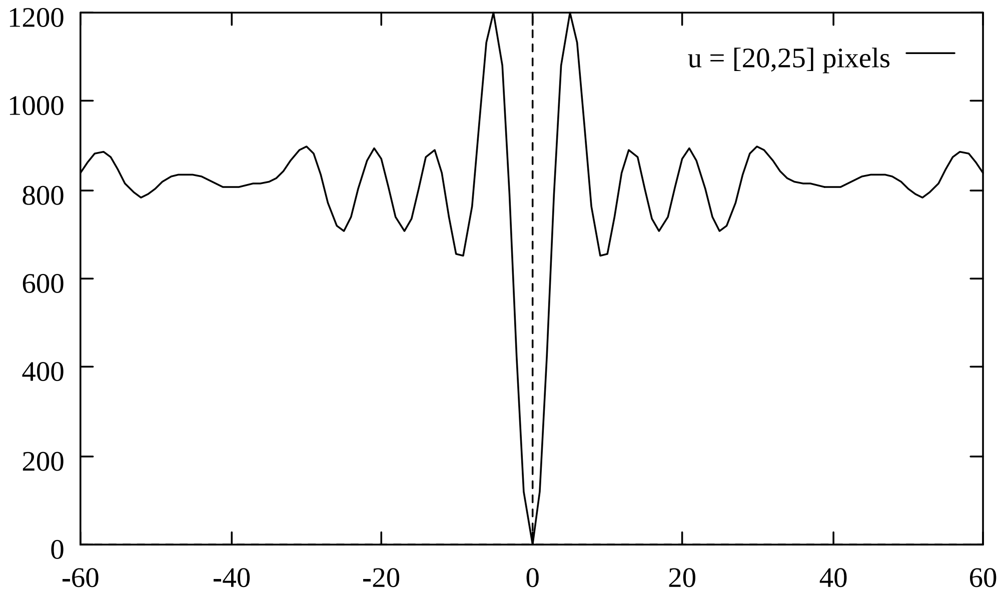

Figure 23: The energy function of the above image. The calculated B.O.A is 10 pixels.

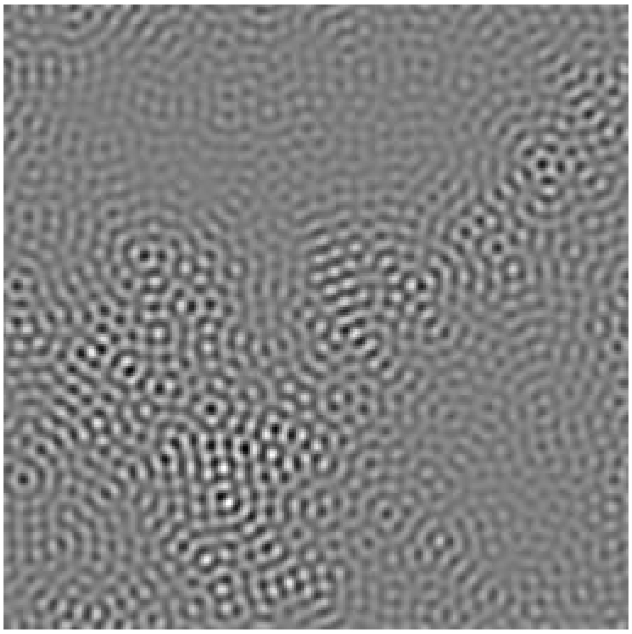

Figure 24: Band pass image with frequencies $25\Delta u \leq u \leq 30\Delta u$.

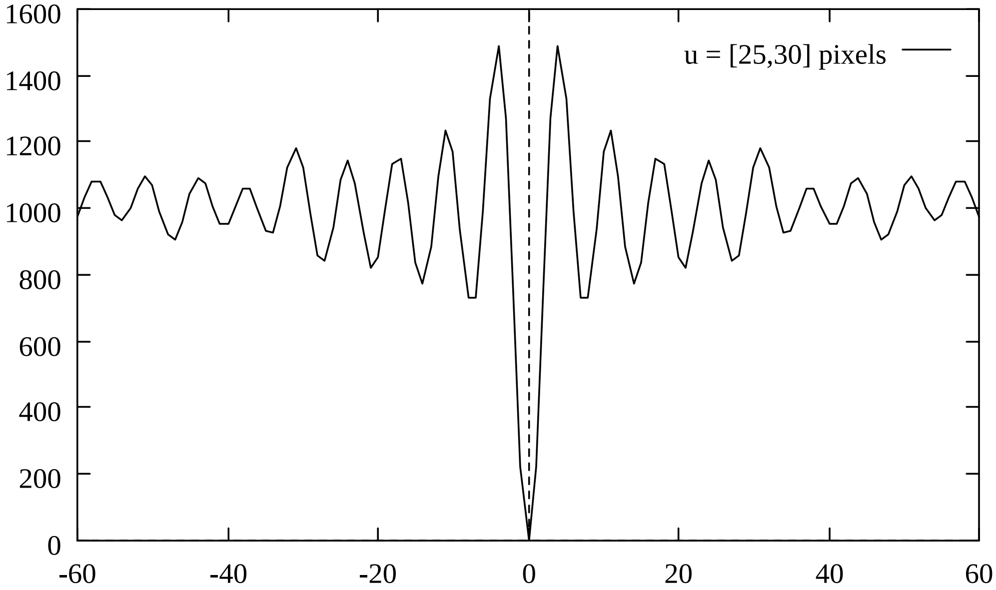

Figure 25: The energy function of the above image. The calculated B.O.A is 8 pixels.

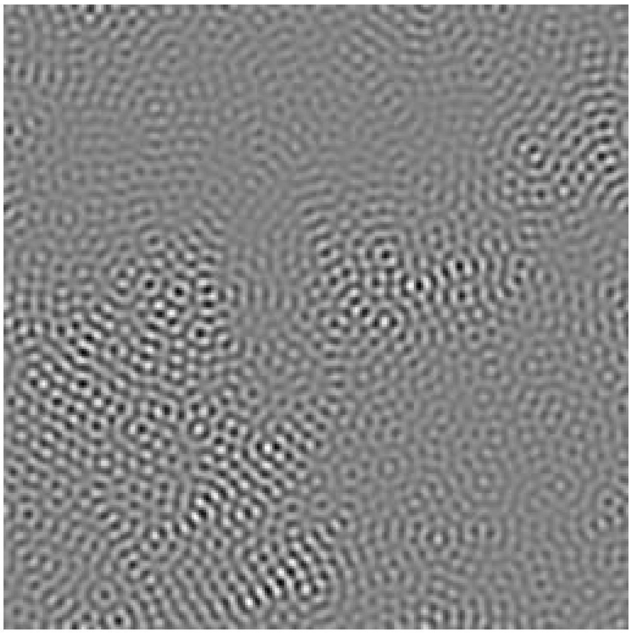

Figure 26: Band pass image with frequencies $30\Delta u \leq u \leq 35\Delta u$.

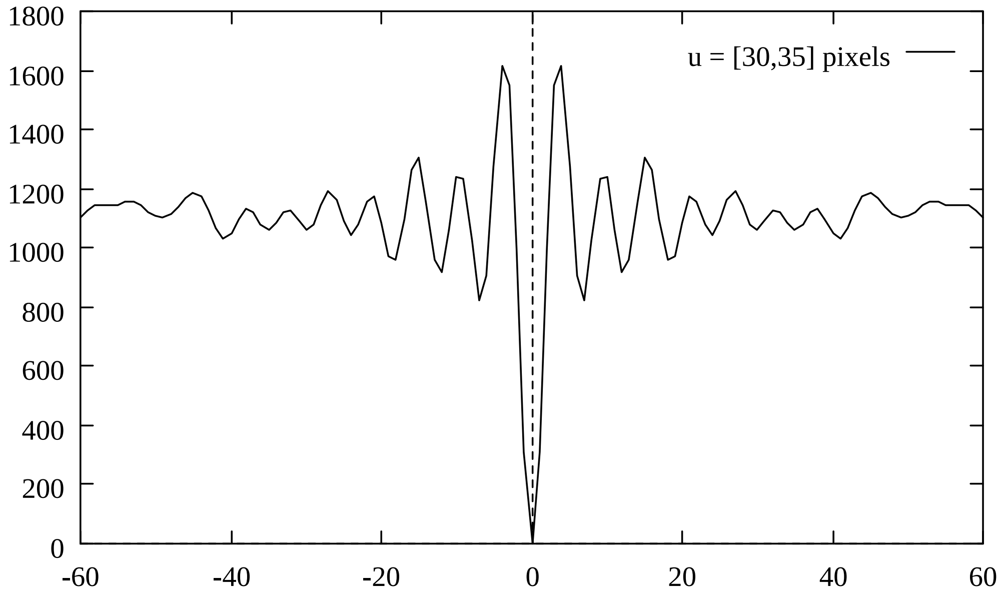

Figure 27: The energy function of the above image. The calculated B.O.A is 8 pixels.

6 Evaluating the Precision of an Estimation

As explained in the previous section the angle θ of the energy line in the frequency domain is equal to the negative speed $-p$, thus, evaluating the angle θ is equivalent to estimating the speed p. However, since our function $I(x,t)$ has a finite support (or it is periodic in space and time) the continuous frequency domain is sampled at a rate of $\Delta u = 1/N\Delta x$ in the u direction and at a rate of $\Delta w = 1/M\Delta t$ in the w direction, where N and M are the number of samples in the spatial and in the temporal domain, respectively. Due to this digitization behavior the precision of the slope estimation is proportional to:

$$(\Delta\theta)^2 \approx (\Delta u^2 + \Delta w^2)/r^2 = \frac{\frac{1}{N^2\Delta x^2} + \frac{1}{M^2\Delta t^2}}{w^2 + u^2}$$

where r is the maximum distance of the energy line from the origin. It can be seen that as u increases (lower levels in the pyramid), the precision of the approximated slope improves. Note that when dealing with a registration with two images then $M = 2$.

Assume we estimate in the higher level of the pyramid a slope q which is close to the real speed p. As stated, the approximated slope is not exact due to the quantization behavior of the transform. If we warp the first image towards the second by our approximated speed, we will still be left with a small displacement between the images which corresponds to a slower speed $p' = (p - q)$. In the frequency domain this effect is expressed by a reduction in the slope of the energy line from $-p$ to $-p'$ (gray line in Figure 10) (see also [7]). This now allows us to take higher frequency images (up to $1/(2p'\Delta t)$) and to improve our estimate.

7 Different Illumination Conditions and Different Sensor Type

An important conclusion that can be made from the previous chapter is that in order to find the motion parameters using any minimization technique, it is worthwhile to start with as lower spatial resolution as possible in order to have a stable initial solution (a broad B.O.A). However, in some cases the above conclusion is wrong. An example for such a case is a set of images which were taken under different illumination condition or by different types of sensors. To illustrate the problem assume that two images such as in Figure 2 were taken under different illumination conditions. To simulate such a case the grayscales of one of the images were multiplied by a factor of 0.5. The energy functions $E_k(p)$ of this case, for 5 levels of the Gaussian pyramid are shown in 28. It is seen that minimization at the lowest resolution (level no. 4) will fail to extract the correct registration parameters (compare these function to the energy functions plotted in Figure 3). The reason for this phenomena is that non-identical illumination conditions cause “smearing” in the low frequencies in the spatio-temporal frequency space. Low pass filters which focus on these frequencies (such as the Gaussian pyramid) magnify this effect and may fail to recover the angle of the energy line (the speed parameter). Band pass filters, on the other hand, such as Laplacian pyramid, may overcome this difficulty. However, using the Laplacian pyramid reduces the stability of the minimization due to narrowing the B.O.A of the energy functions. Energy functions for 5 levels in the Laplacian are plotted in Figure 29.

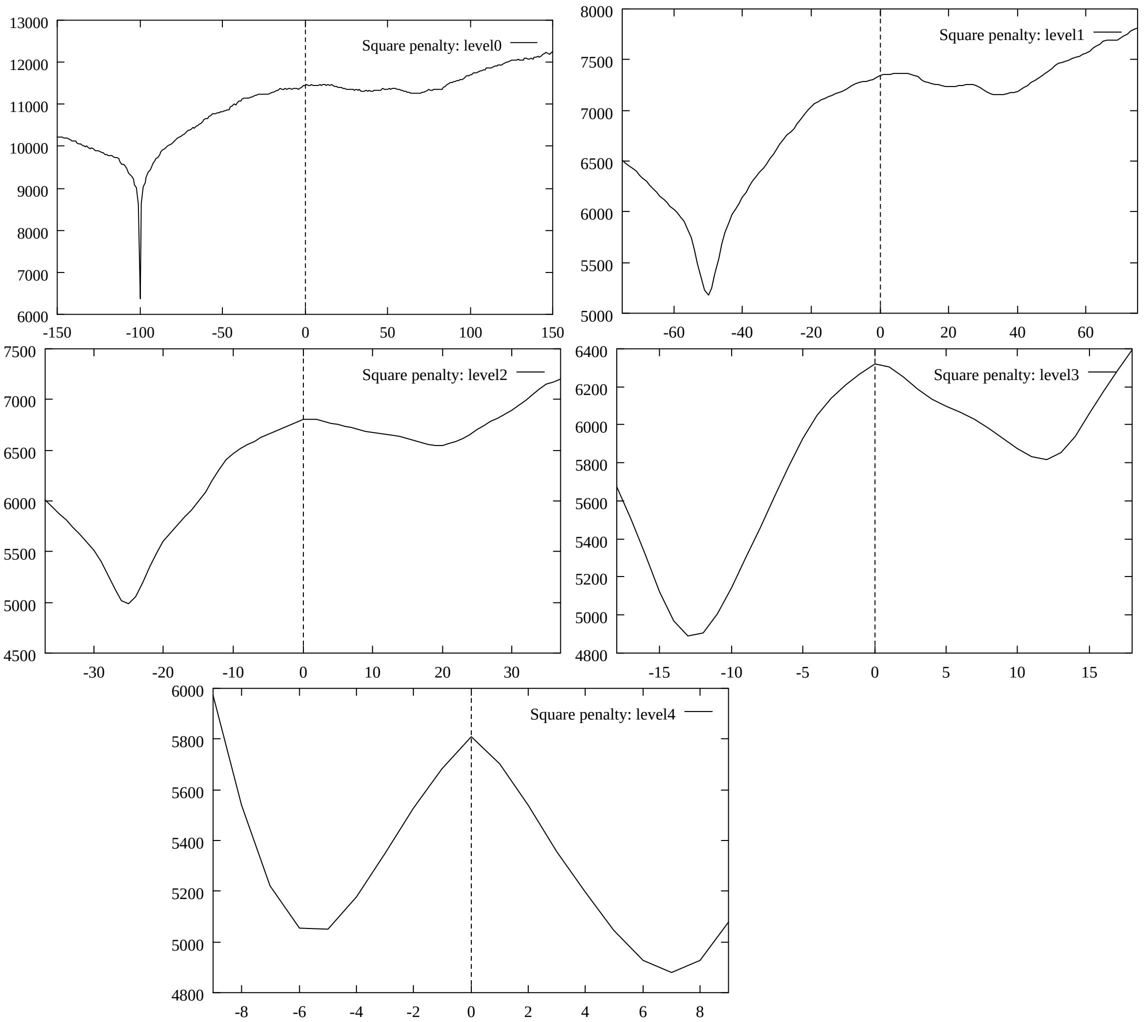

Figure 28: Different illumination condition: the energy function for 5 levels in the Gaussian Pyramid. The highest level fails to extract the global minimum.

A more complex case exists when the grayscales of the two images are different due to different types of sensors used to acquire these images. In this case the corresponding grayscales of the images can be totally different and even non-monotonic. The fact that the images show the same scene is expressed only in the higher frequencies of the images, therefore, a comparison can be made only at these frequency levels. However, using the high frequencies will drastically reduce the stability of the convergence and the multiresolution scheme will not produce a better performance. An example can be seen in the following simple case. In Figure 30 two identical images were covered by a non-monotonic grayscale. Trying to register

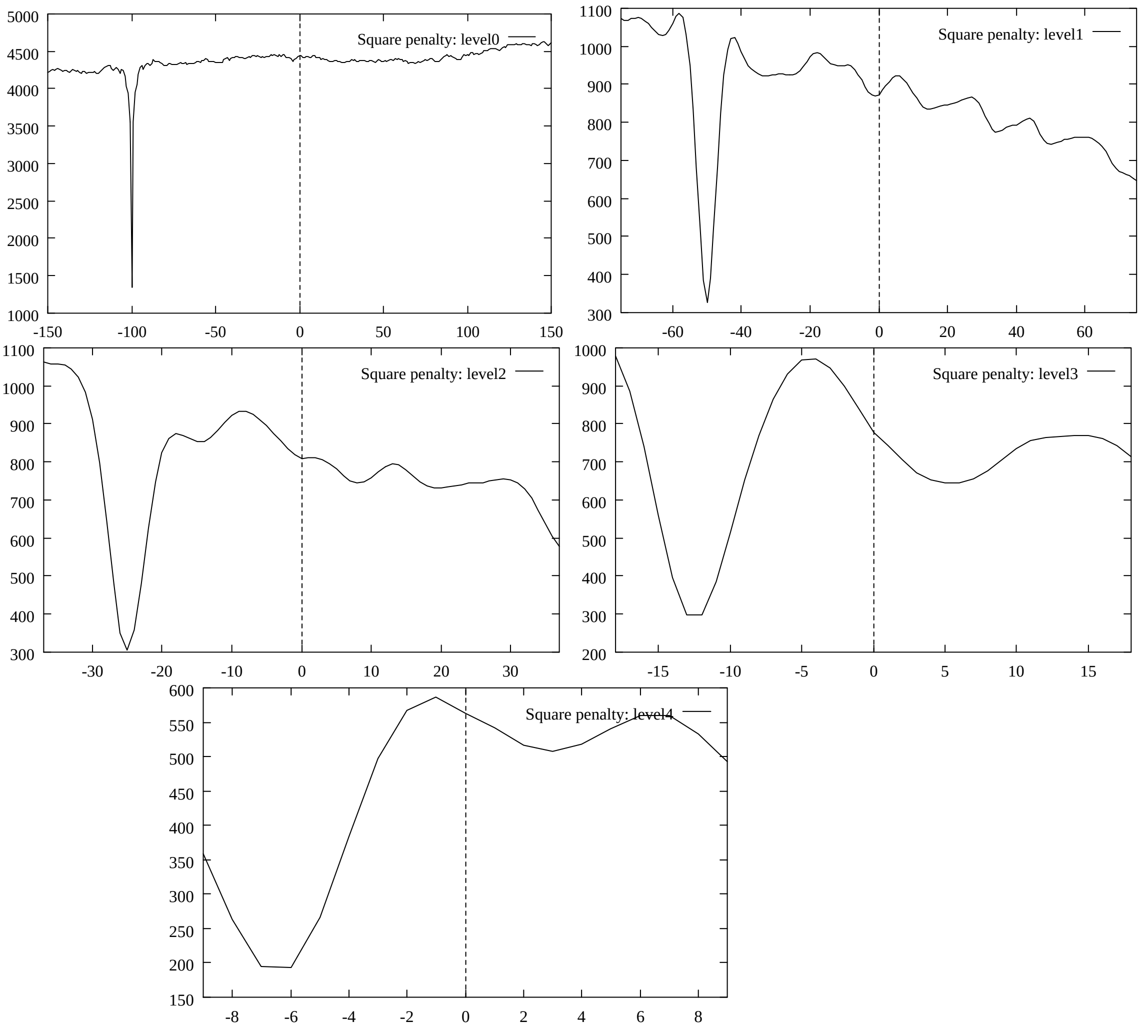

Figure 29: Different illumination condition: the energy function for 5 levels in the Laplacian Pyramid. The highest level enables extraction of the global minimum.

these two images by minimizing $E(p)$ of the original images is hopeless. Trying to do so by the edge information of these images is a better possibility. Figure 31 is the edge image of Figures 30. It is produced by applying an edge detector on one of the original images. The Energy functions $E_k(p)$ for 4 levels in the Gaussian pyramid of the edge images are plotted in Figure 32. It can be seen from these energy functions that the real speed parameter can be found but the B.O.A is quite narrow and it does not increase significantly at higher levels of the pyramid. This results in a limited performance of the energy minimization scheme.

In order to improve the registration performance when the images are taken from different sensors it is possible to use the following approach:

- In order to reduce aliasing effects as explained above it is possible to use additional

Figure 30: Two images with non-monotonic grey scales.

filtering such as directional selective filtering in the spatial and temporal frequency domain which reduces the chances of accumulating aliasing from other areas in the frequency domain, or use non linear filtering such as a voting scheme etc. (see also [1, 8, 16]).

- The inconsistencies in the lower frequencies of the pictures can be bypassed by performing the registration based on depth information obtainable from the images. Depth information can be deduced if a set of images are available for each sensor type. The depth information is consistent between sensor types.

8 Conclusions

In this report, the difficulties of the registration problem were discussed and demonstrated, especially for the case when the images to be registered are taken from different types of sensors. The advantages and limitations of the multiresolution technique were detailed and formulated for the case of translatory transformations. In order to bring the multi-sensor registration into acceptable performance ranges, research should continue in several directions:

- Classical L.S. estimation of the registration parameters (square penalty) is influenced by the aliasing effect which can be considered as outlier measurements. In order to reduce the influence of the outliers, robust estimation should be applied to this problem. Using robust estimation techniques, outliers are iteratively filtered out of the measurements.

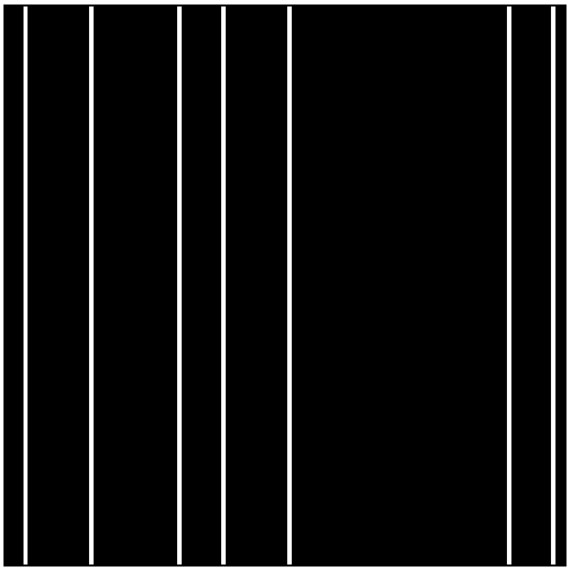

Figure 31: An edge image of Figure 30.

- In order to isolate the aliasing energy from the real energy it is essential to use multi-band filtering where filtering is done both in the spatial and in the temporal domains. This kind of filtering can help us distinguish between aliasing and real energy by using a voting scheme or by applying some line detection in the frequency domain.

- In dealing with different sensor types, depth information may be used in addition to the grayscale information to assist in registration. This can be done if a stream of images is available from each sensor. Note that it is not necessary to recover the full depth information to perform registration. Partial depth information, such as depth ordering which may be easier to obtained, is sufficient.

- Additional directions which would be a continues extension to the presented study is to formulate the limitations and performance of the multiresolution scheme for more complicated motion models, such as affine and perspective transformations (in the image plane). This study can lead to a better understanding of the multiresolution technique which may be dependent on the transformation type we seek.

References

[1] E.H. Adelson and J.R. Bergen. The extraction of spatio-temporal energy in human and machine vision. In *Proc. Workshop on Motion*, 1986.

[2] G. Adiv. Determining three-dimensional motion and structure rom optical flow generated by several moving objects. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 7:384–401, 1985.

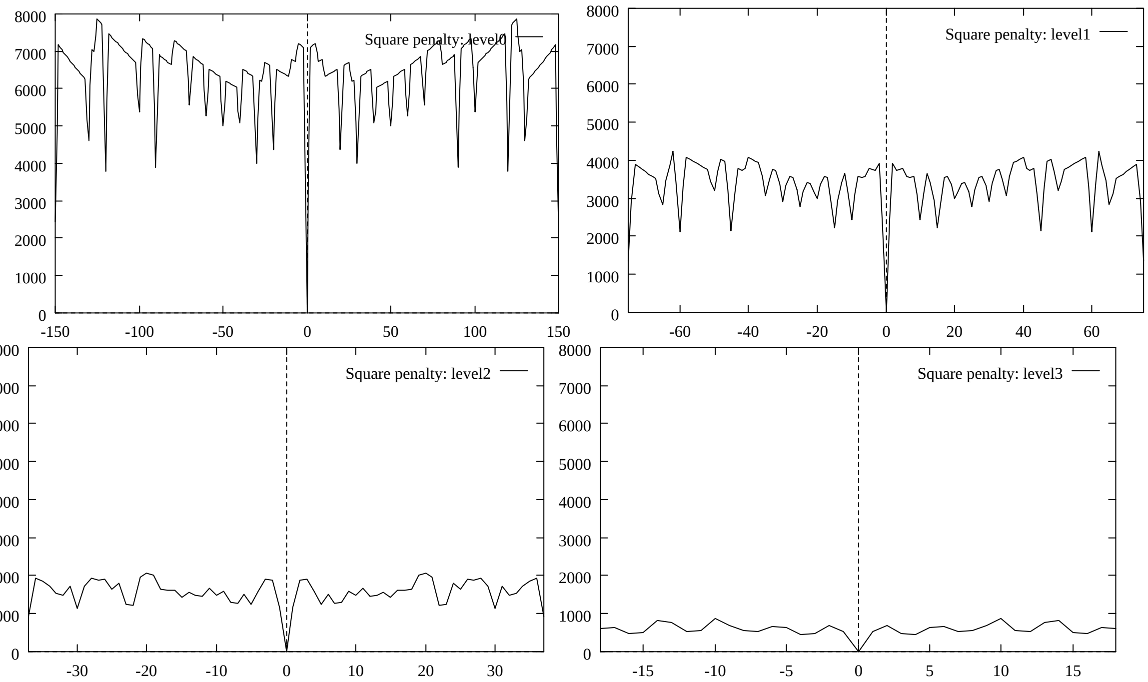

Figure 32: Edge images: the energy functions for 4 levels in the Gaussian pyramid. The B.O.A is relatively narrow at all of the levels.

[3] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2:283–310, 1989.

[4] J.R. Bergen, P. Anandan, K.J. Hanna, and R. Hignorani. Hierarchical model-based motion estimation. In *European Conference on Computer Vision*, 1992.

[5] P.J. Burt. Smart sensing within pyramid vision machine. *Proceedings of the IEEE*, 76, 1988.

[6] P.J. Burt and E.H. Adelson. The laplacian pyramid as a compact image coding. *IEEE Transactions on Communications*, 31, 1983.

[7] P.J. Burt, R. Hignorani, and R.J. Kolczynski. Mechanisms for isolating components patterns in the sequential analysis of multiple motion. In *Proc. Workshop on Motion*, 1991.

[8] David Heeger. Model for the extraction of image flow. *Journal of the Optical Society of America*, 4, 1987.

[9] B.K.P. Horn. *Robot Vision.* MIT Press, 1986.

[10] B.K.P. Horn. Relative orientation. *International Journal of Computer Vision*, 4:59–78, 1990.

[11] B.K.P. Horn and B.G. Schunk. Determining optical flow. *Artificial Intelligence*, 17:185–204, 1981.

[12] M. Irani, B. Rousso, and S. Peleg. Detecting and tracking multiple moving objects using temporal integration. In *European Conference on Computer Vision*, 1992.

[13] K. Kanatani. *Group-Theoretical Methods in Image Understanding*. Springer-Verlag, 1990.

[14] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theort. *Nature*, 317, 1985.

[15] D. Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8, 1986.

[16] A.B. Watson and A.J. Ahumada. Model of human visual-motion sensing. *Journal of the Optical Society of America*, 2, 1985.