Studies in Correlation 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. Correlation based registration relates to those techniques which find the warping parameters from the greyscale differences between the two images. The main advantage of these kind of techniques is that a correspondence between features in the images is not required (finding the correspondence is exponential in the number of features). However, the performance of correlation based algorithms will be adequate only in the cases where the difference in the viewing position is relatively small and the illumination conditions in the two images are similar .

2 The Parameters to be Found

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:

$$I_1(x+u, y+v) = I_2(x, y) \quad .$$

i.e. the two images are greyscale 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 now appears at the location $(x+u, y+v)$. If there are no constraints on (u, v) then for each pixel we have:

$$I_1(x+u(x,y), y+v(x,y)) = I_2(x, y)$$

and we obtain an under-determined system of equations; n equations and $2n$ unknowns. This is a well known problem called the "aperture problem" in which we have infinite solutions for any input. In order to make the system solvable we must constrain (u, v) according to some apriori knowledge, intrinsic to the specific problem at hand.

2.1 Local Parametric v.s. Global Parametric Motion Model

A possible constraint to be imposed on the solution comes from the fact that the depth of a regular scene does not change rapidly and therefore the retinotopic motion - $u(x,y)$ and $v(x,y)$ are "smooth" functions. This is known as the *regularization* constraint. In this case $u(x,y)$ and $v(x,y)$ are replaced by unknown parameters $u_{x,y}$ and $v_{x,y}$ describing the local motion of a small region centered at (x,y). This kind of motion description is called "local parametric motion model".

An alternative possibility is to try to describe the retinotopic motion by fewer parameters. This can be done if some assumption is made about the scene structure. Possible assumptions that are widely used are:

- The scene is rigid.
- Known structure, i.e. the depth of each point in I_1 is known.
- Known parametric structure. For example, a planar scene can be described by the equation $Z = aX + bY + c$ where a, b and c are unknown parameters.

In these cases $u = u(x,y,\mathbf{p})$ and $v = v(x,y,\mathbf{p})$ where $\mathbf{p}$ is a parameter vector to be found. Since $\mathbf{p}$ describes the entire retinotopic motion, this kind of motion description is called "global parametric motion model". The assumptions listed above are very relevant to our project and therefore I will concentrate on this motion model.

3 Global Parametric Motion Model

3.1 The Rigidty Constraint

Assume a 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 motion of a 3D point $\mathbf{P} = (\mathbf{X}, \mathbf{Y}, \mathbf{Z})^{\mathbf{t}}$ can be described in the camera coordinate frame by:

$$\frac{\partial \mathbf{P}}{\partial t} = -\mathbf{T} - \mathbf{R} \times \mathbf{P}$$

Expanding this into components yields:

$$\begin{aligned} \frac{\partial X}{\partial t} &= -T_z - R_y Z + R_y Y \\ \frac{\partial Y}{\partial t} &= -T_y - R_z X + R_x Z \\ \frac{\partial Z}{\partial t} &= -T_z - R_x Y + R_y X \end{aligned} \tag{1}$$

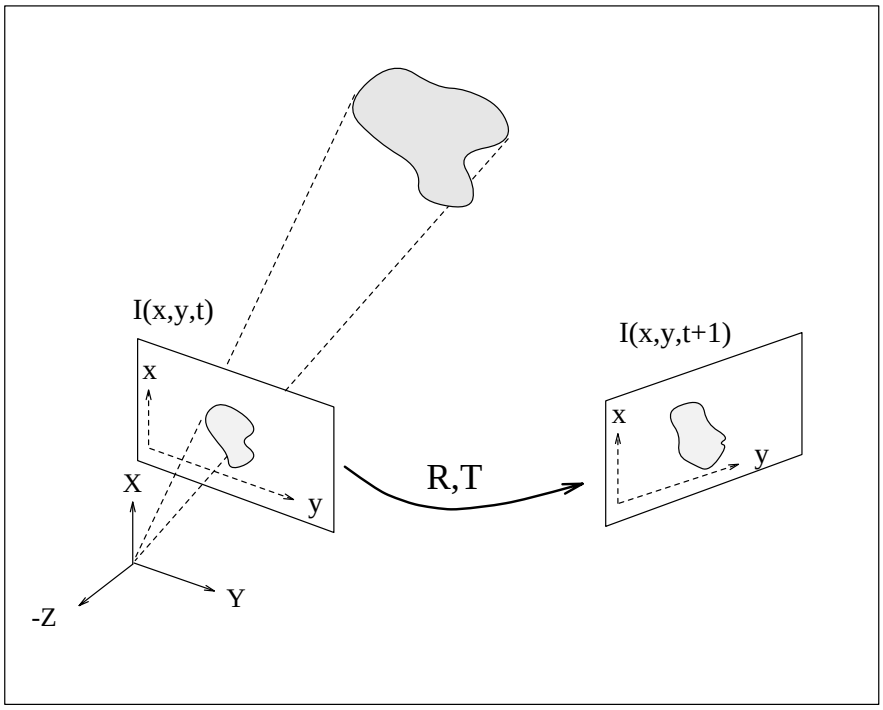

Figure 1: Two images taken from a moving camera.

Projecting **P** onto the image plane gives:

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

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

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

Substituting the derivatives from equation 1 yields:

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

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. It can be seen from this equation that the rigidity constraint reduces the number of parameters to be found to $6 + n$ unknowns; n depth parameters $Z(x, y)$, and 6 motion parameters determining $\mathbf{T}$ and $\mathbf{R}$.
Note:

- There is a degree of freedom in the magnitude of the translation and depth, so 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 Structure

In the case where the depth $Z(x, y)$ is known for every pixel located at (x, y):

$$(u, v)^t = A'T + BR$$

where $A' = A/Z(x, y)$ is a known matrix. In this case there are only 6 unknowns to be found - the motion parameters.

3.3 Parametric Structure

Since our project deals mainly with planar scenes, the following discussion will be concentrate on this parametric structure.

3.3.1 A Plane Under Perspective Projection

If the scene is planar, all points of this plane satisfy the equation: $Z = a + bX + cY$.
Dividing 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:

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

In this motion model we have 9 unknowns to be found. Since $(\alpha + \beta + \gamma)$ 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 coupled parameters which can be written as:

$$\begin{aligned} u &= a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 xy \\ v &= a_6 + a_7 x + a_8 y + a_4 xy + a_5 y^2 \end{aligned}$$

This motion model is sometimes called "Pseudo Projective Model".

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:

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

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:

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

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

4 Multiresolution Estimation

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

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

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

$$E = \sum_{x,y} J(I_1(x + u(x, y, \mathbf{p}), y + v(x, y, \mathbf{p})) - I_2(x, y))$$

where J is some penalty function. Since I_1 depends non-linearly on $\mathbf{p}$ we have a non-linear minimization problem at hand.
We have to choose:

- The penalty function
- The minimization technique to use.

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 following example two images were displaced from each other by 100 pixels in the x direction (see Figure 2). The energy function (square penalty) as a function of p is shown in Figure 3:

$$E(p) = (I_1(x+p, y) - I_2(x, y))^2 \tag{2}$$

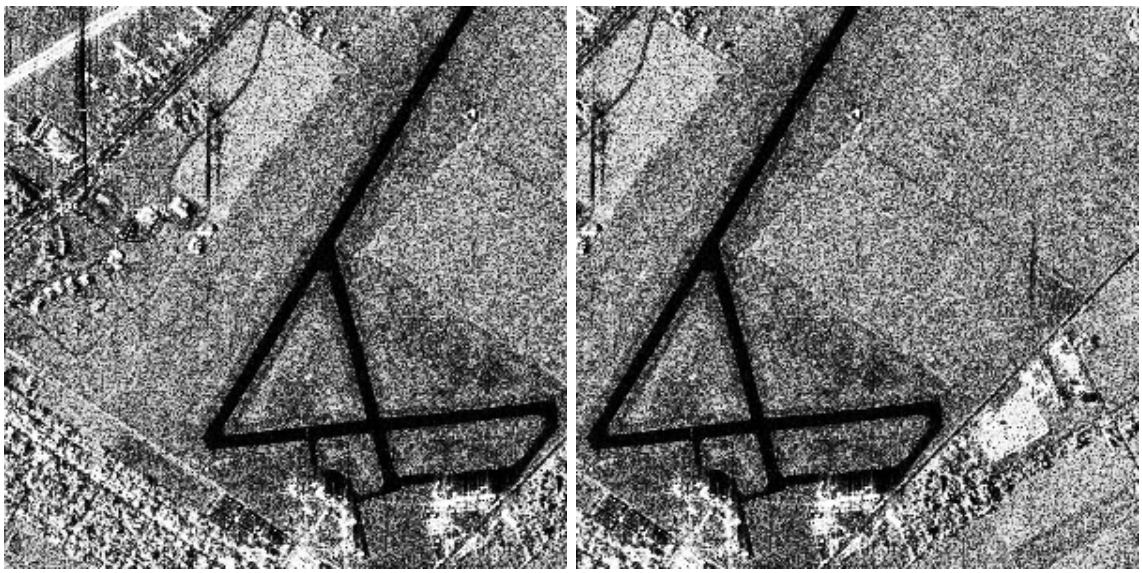

Figure 2: Two images translated from each other by 100 pixels.

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 (exponentially with the number of parameters). Therefore we wish to expand the basin of attraction and to use some gradient minimization from sparse samples of the parameter domain. 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 $k-1$ by the following manner:

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

The convolution with G is a low pass filter and S is a sampling with a "comb" function (in general, having a sampling rate of 1:2).

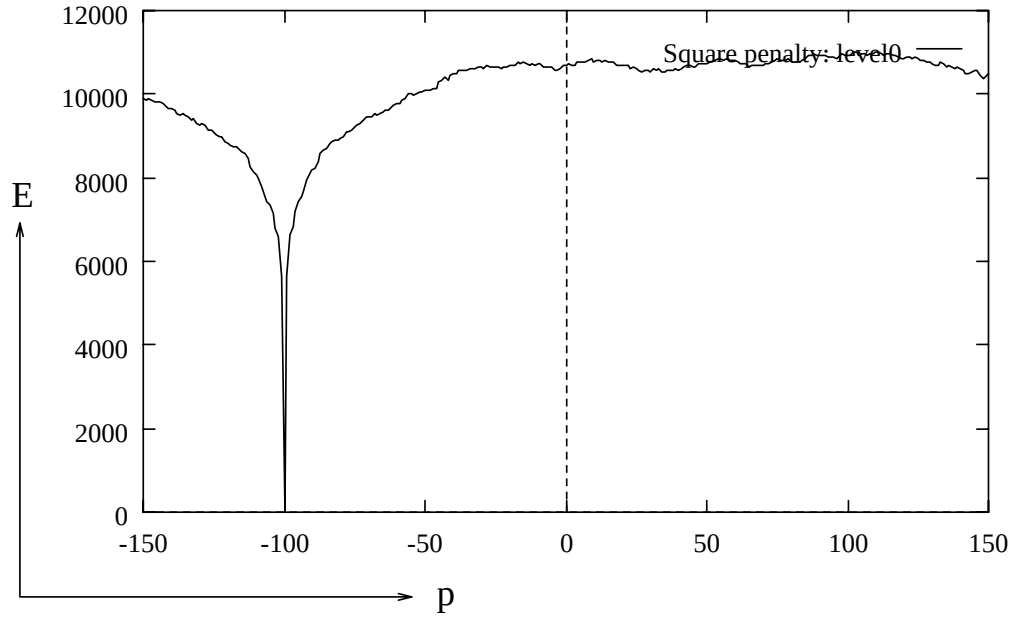

Figure 3: Energy function relative to the translation parameter p.

Two types of pyramid are commonly used: the *Gaussian* pyramid and the *Laplacian* pyramid. In the Gaussian pyramid we generate the next level by convolving the current level with a Gaussian kernel followed by subsampling. This is equivalent to a low pass filtering. The s.t.d. 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 the frequency above $n\Delta u/2$. 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 $\sigma_x = \frac{4}{n\Delta u} = \frac{4n\Delta x}{n} = 4\Delta x$

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. 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 levels:

$$E_k(p) = (I_1^k(x+p, y) - I_2^k(x, y))^2$$

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.

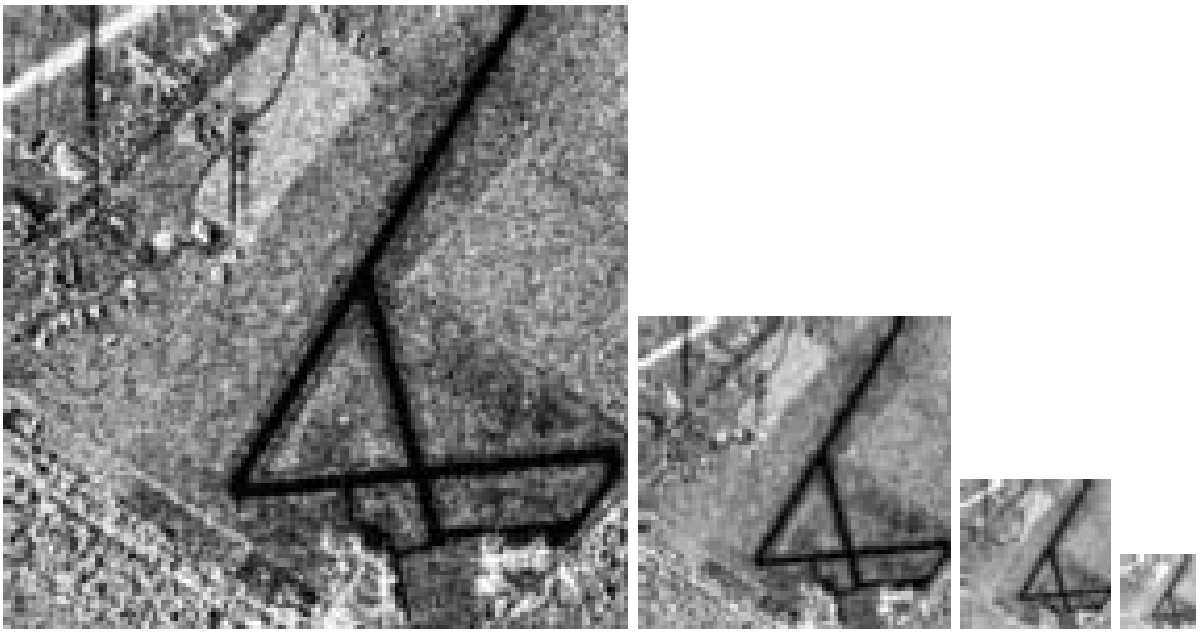

Figure 4: Gaussian Pyramid with 4 levels.

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 precision in the localization of the global minimum p_{min} decreases.

Using this behavior the motion estimation between two images I_1 and I_2 proceeds 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- converge to the minimum from the given initial guess.
5. Go to step 3 until you arrive to level 0.

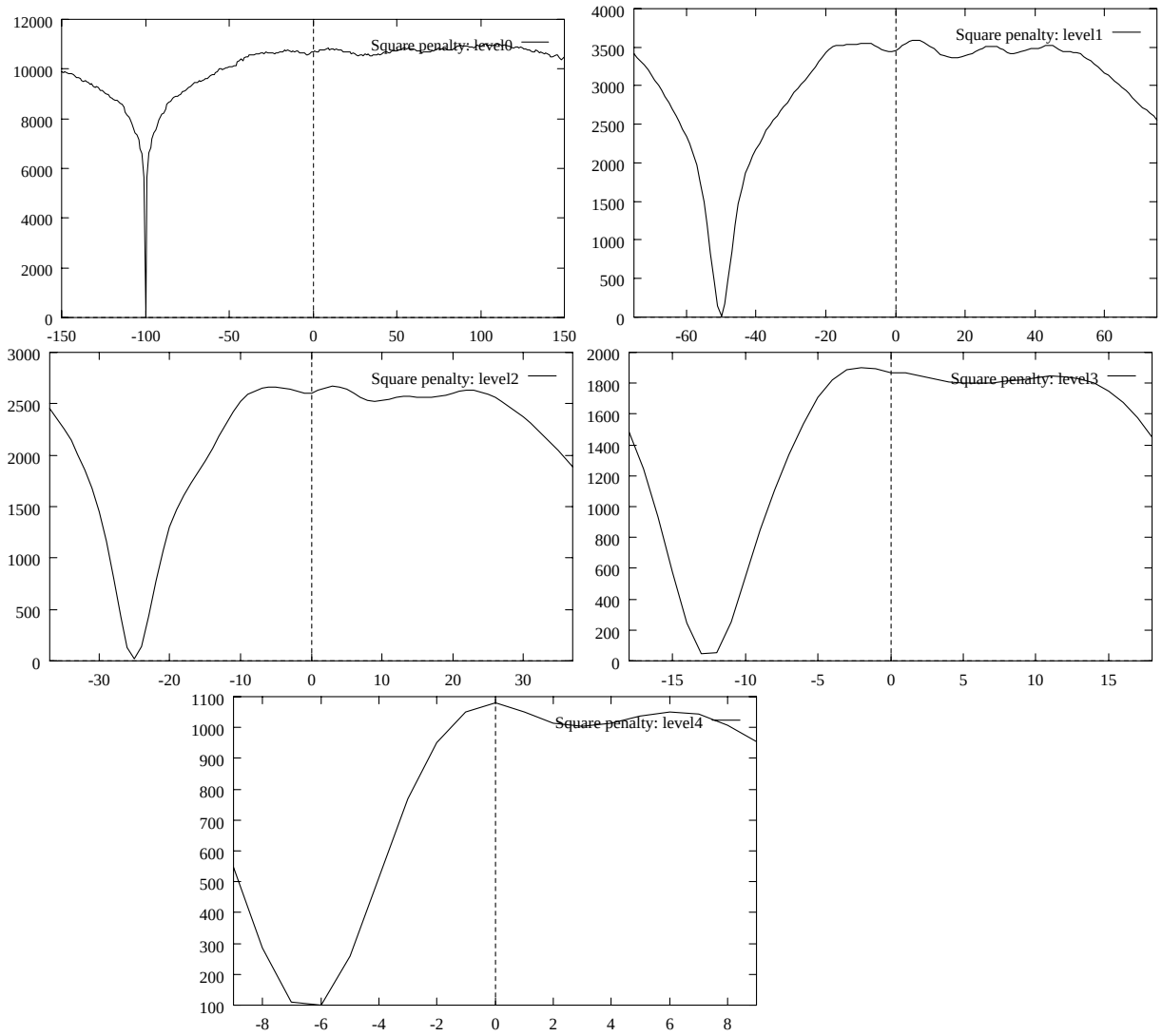

Figure 5: Five energy functions for five pyramid levels.

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 Figures 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. It is demonstrated that the algorithm finds the correct transformations in all these cases.

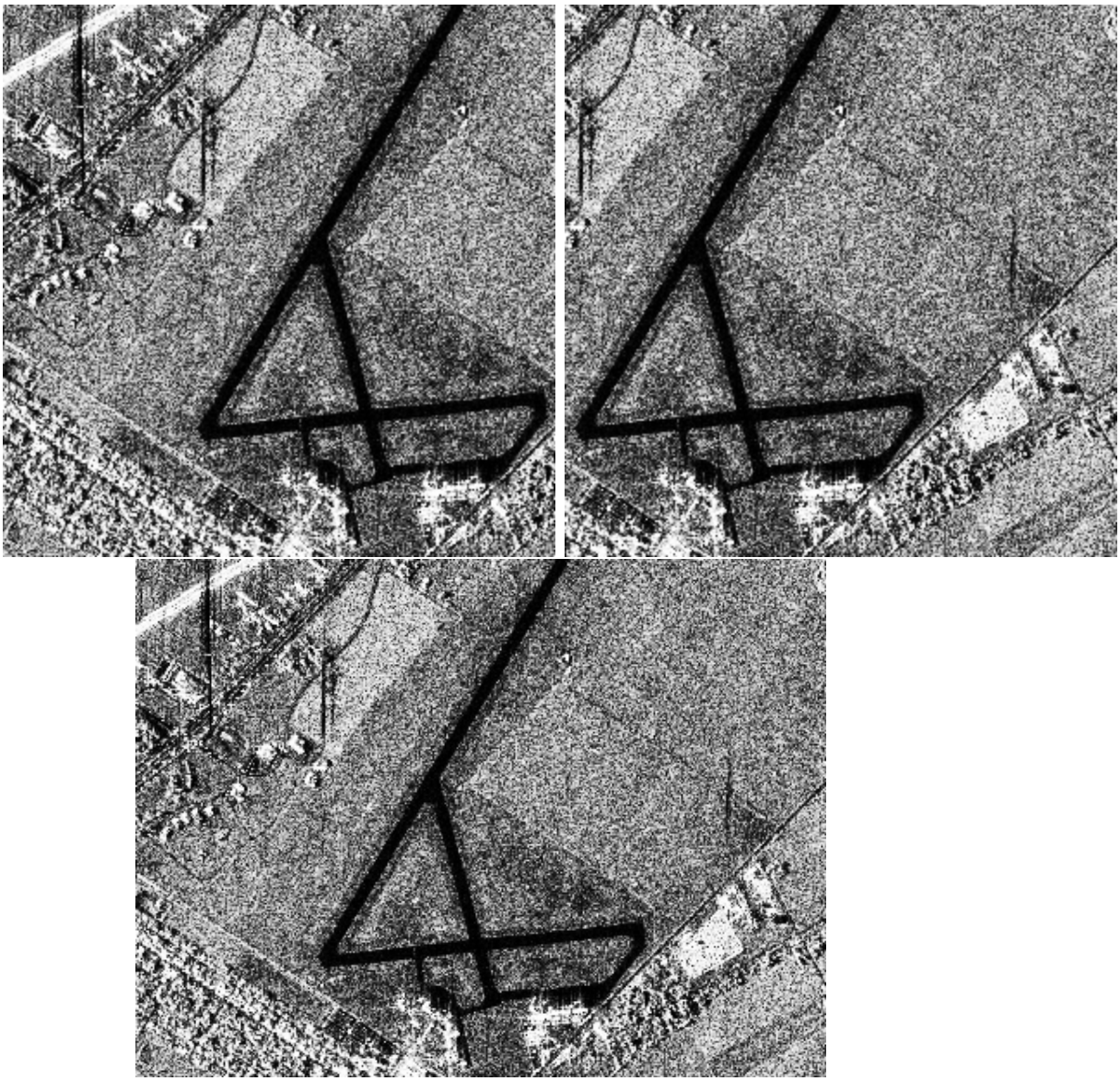

Figure 6: Registration for one dimensional translation.

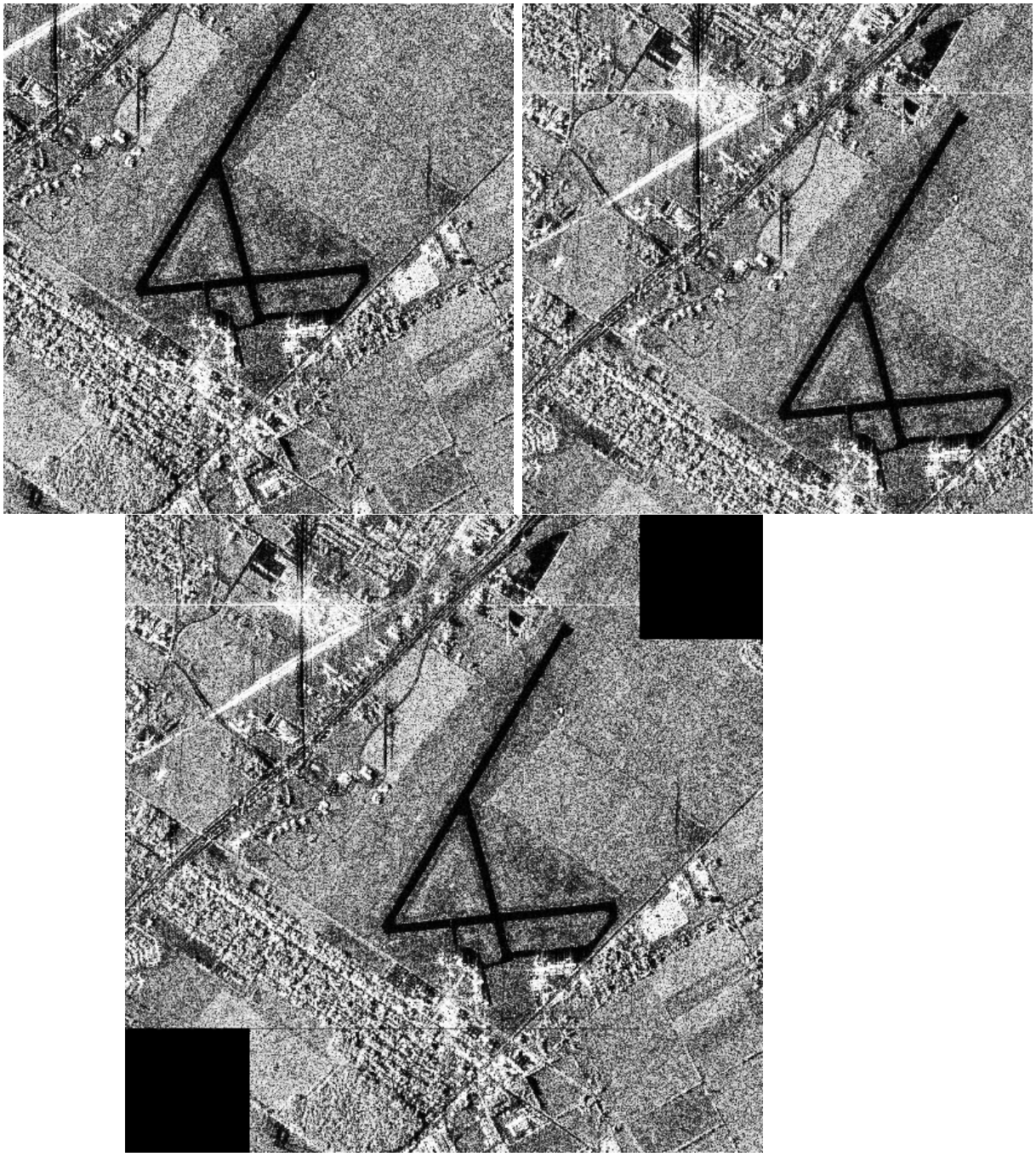

Figure 7: Registration for two dimensional translation.

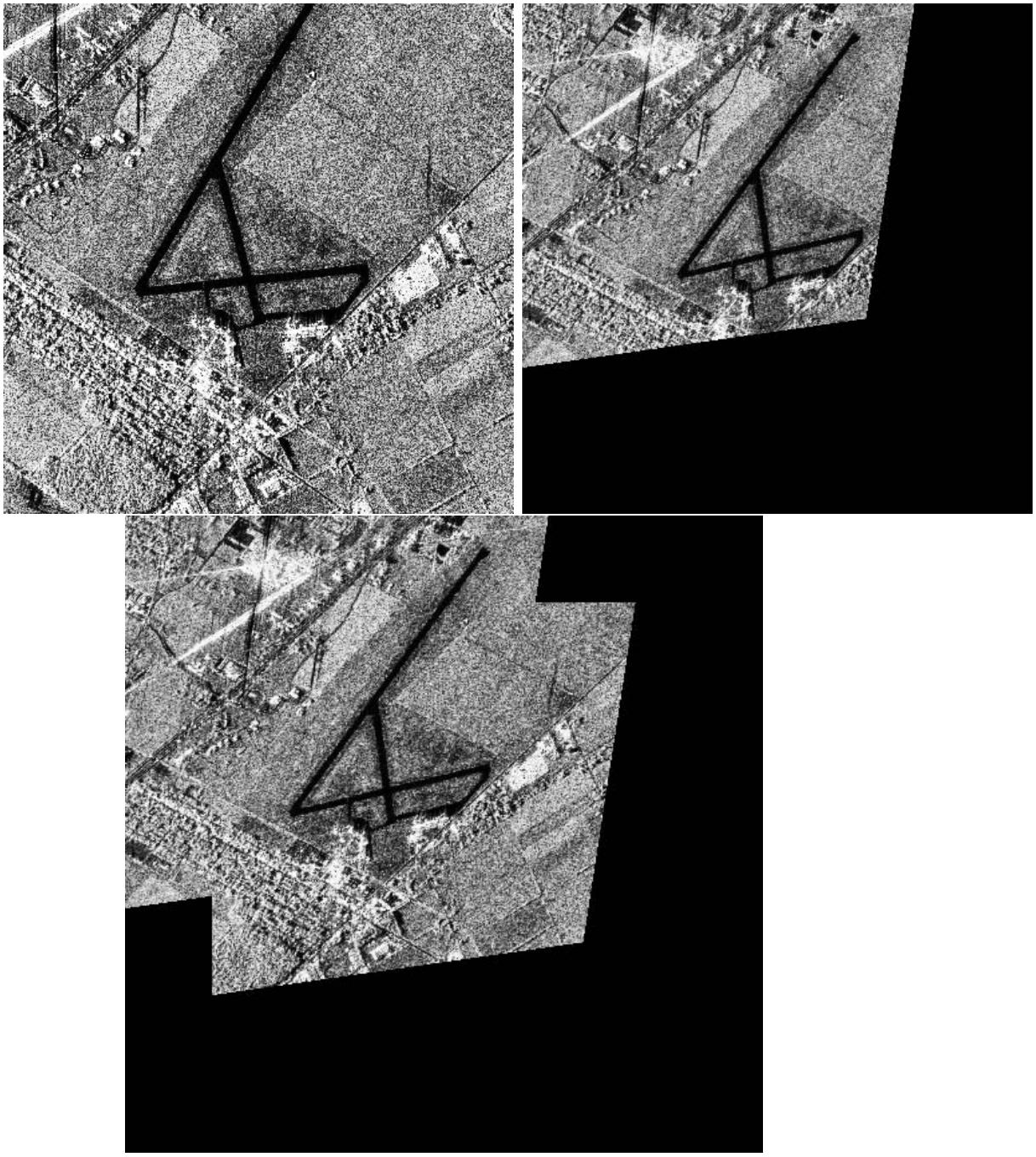

Figure 8: Registration for affine tranformation.

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 order to investigate this question assume a simple example where a one dimensional function is translated at a constant velocity:

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

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).

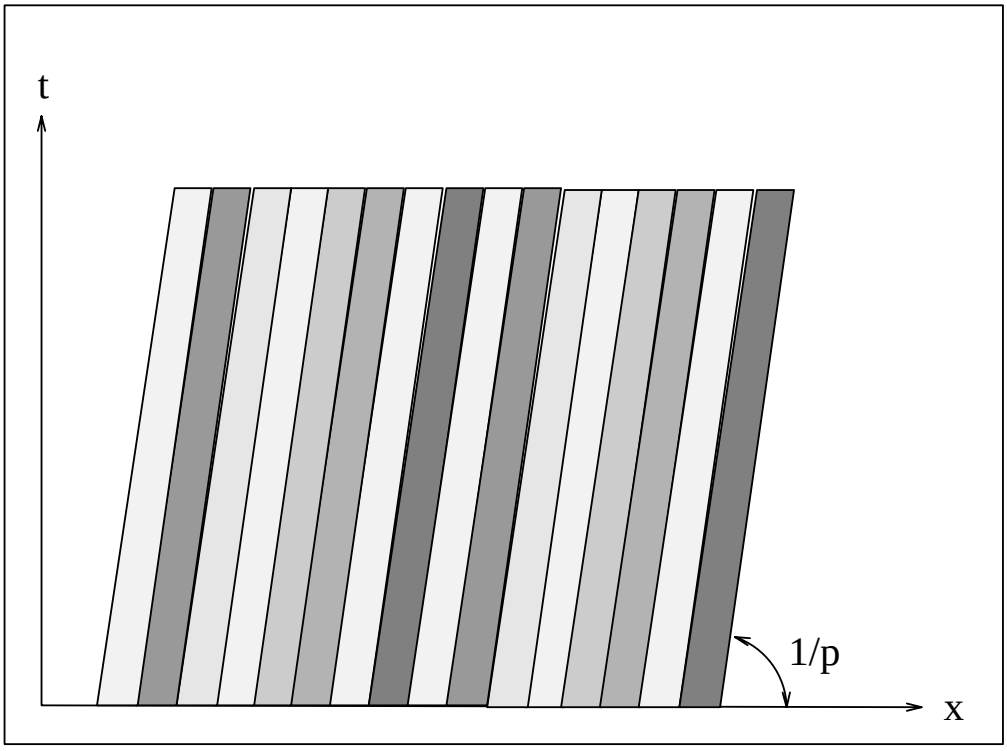

Figure 9: spatio-temporal representation of a translated one dimensional image.

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

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

$$
\begin{aligned}
&= \int \hat{I}_0(u) \exp^{-2\pi i u p t} \exp^{-2\pi i w t} \, dt \\
&= \hat{I}_0(u) \int \exp^{-2\pi i t(w+up)} \, dt \\
&= I_0(u)\delta(w + up)
\end{aligned}
$$

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 with its magnitude to the velocity value. This is demonstrated by the black line in Figure 10.

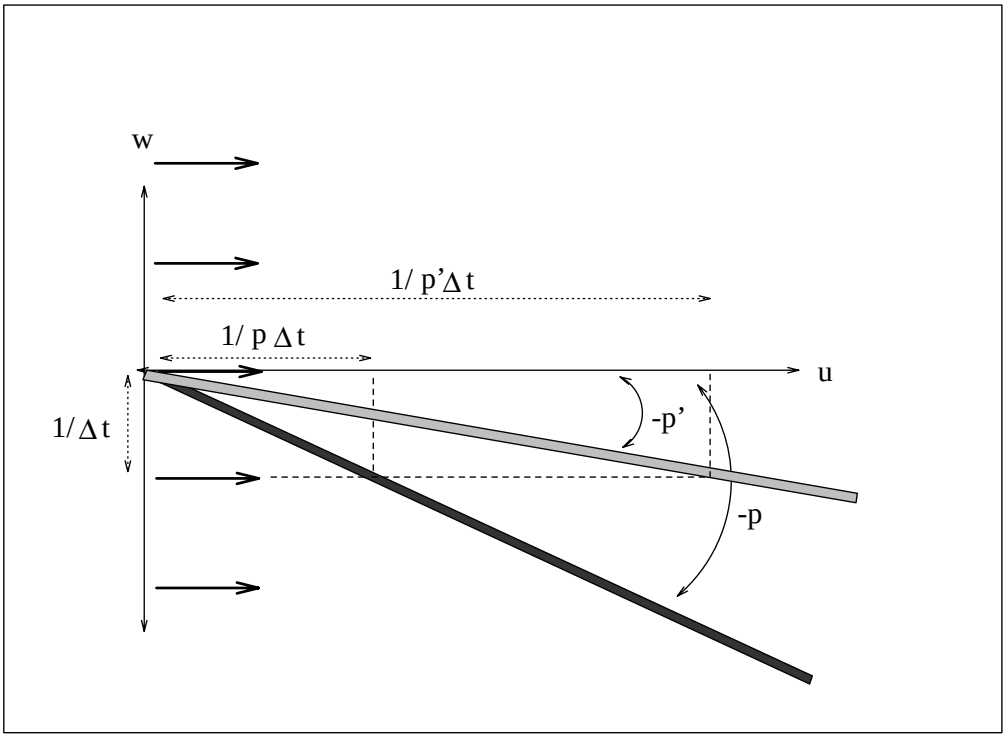

Figure 10: A moving image is represented by a slanted line in the frequency spatio-temporal domain, with a negative slope equal to the speed. The values at this line are equal to the static Fourier values $I_0(u)$.

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

$$E(v) = \int \int [I(x,t) - I(x+v, t+1)]^2 dx dt \tag{3}$$

Expanding $I(x+v, t+1)$ around (x,t) and taking the first order approximation yields:

$$I(x+v, t+1) = I(x,t) + vI_x + I_t \tag{4}$$

where

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

Substituting Equation 4 into Equation 3 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 motion speed), we get:

$$E(v) = \int\int (v-p)^2 u^2 |\hat{I}(u,w)|^2 dudw \tag{5}$$

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

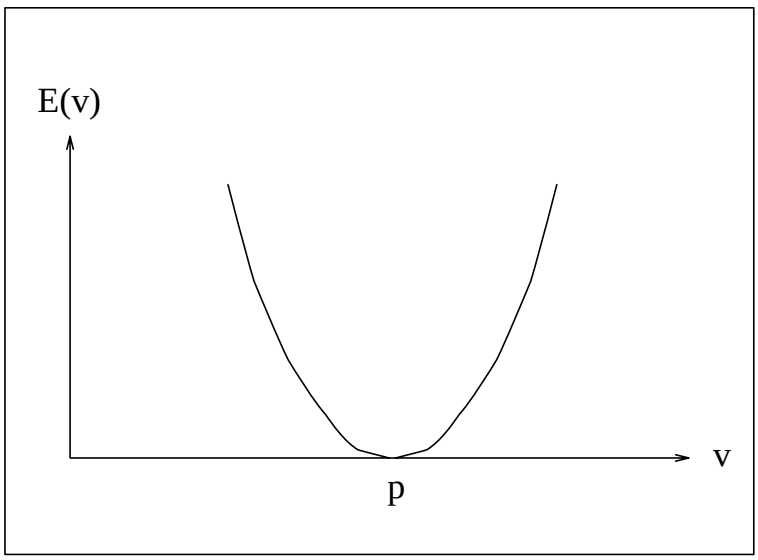

Figure 11: The energy function $E(v)$ as a function of v.

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 replicas 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 and will deteriorate the energy function (see Figure 12). Simple geometry calculations shows that

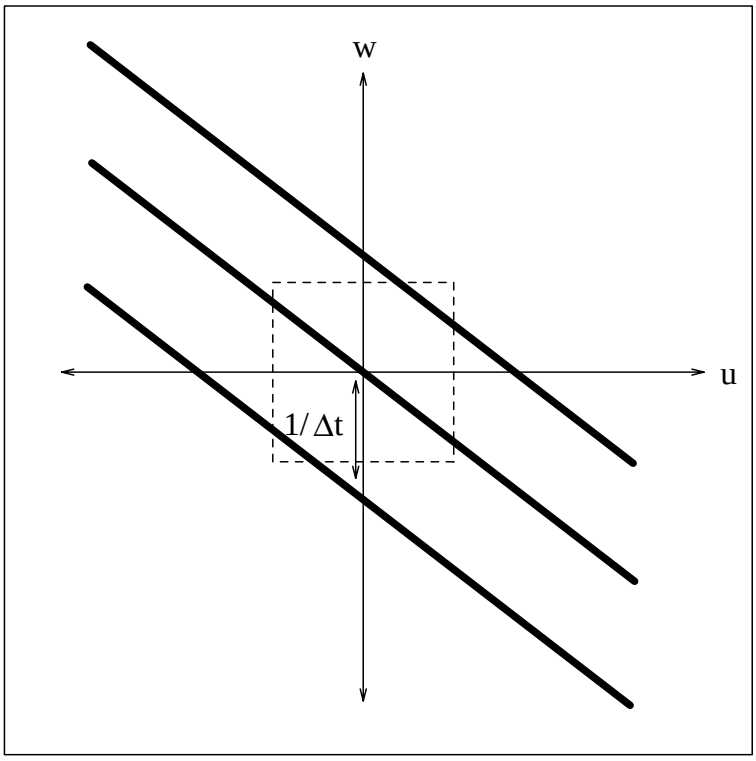

Figure 12: The frequency domain when sampling $I(x,t)$ every Δt.

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:

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

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 2) 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 apriori informstion 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 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. up to 1 pixel. Taking that 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.

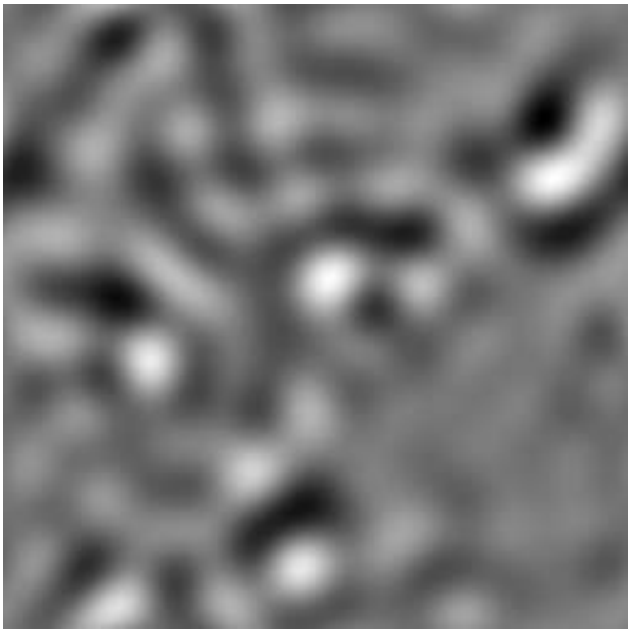

Figure 14: Band pass image with frequencies $0\Delta u \leq u \leq 5\Delta u$.

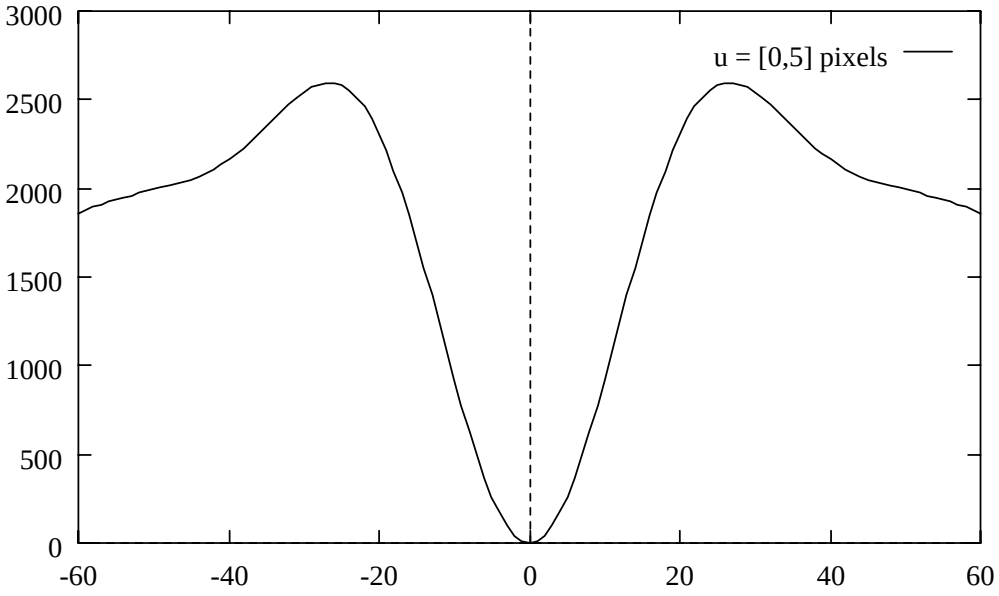

Figure 15: The energy function of the above image. The calculated B.O.A is 54.

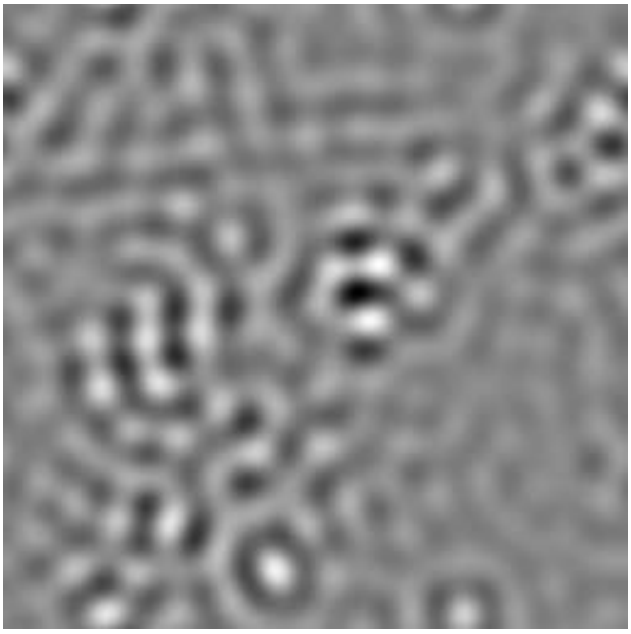

Figure 16: Band pass image with frequencies $5\Delta u \leq u \leq 10\Delta u$.

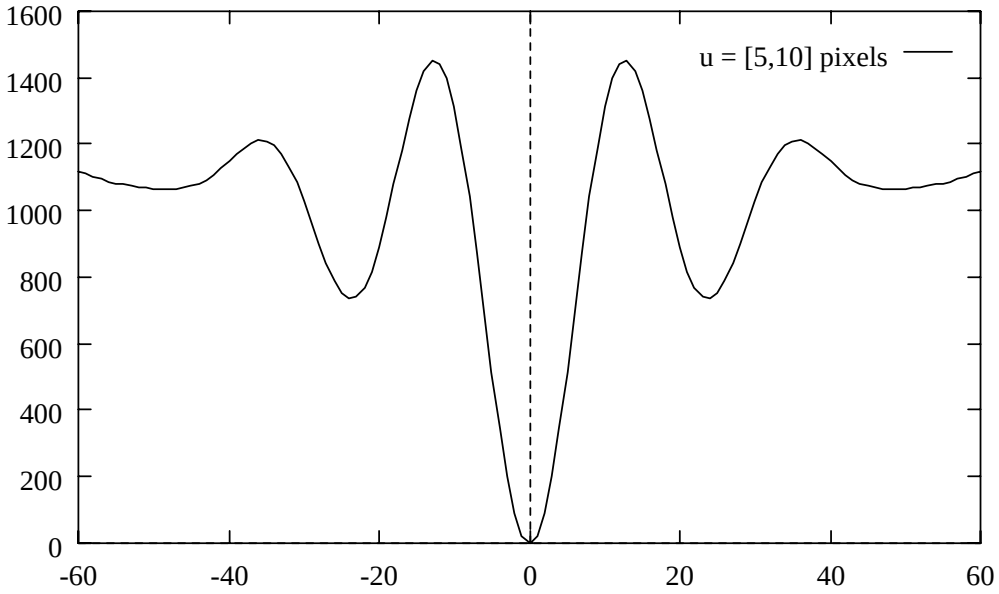

Figure 17: The energy function of the above image. The calculated B.O.A is 26 pixels.

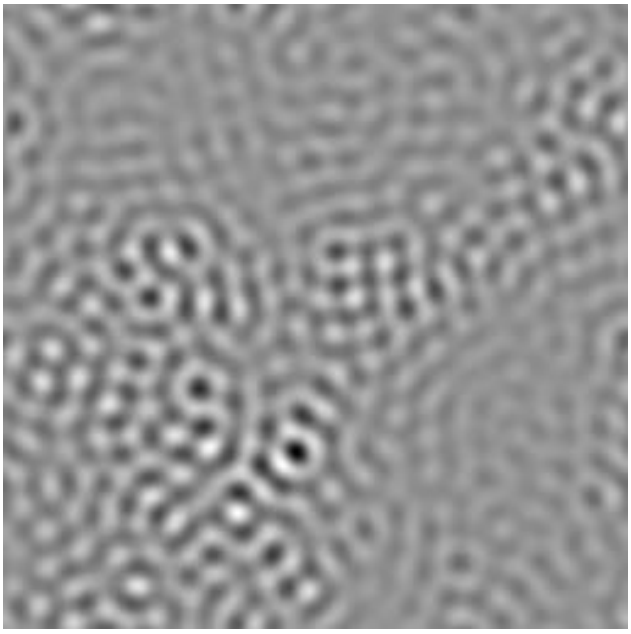

Figure 18: Band pass image with frequencies $10\Delta u \leq u \leq 15\Delta u$.

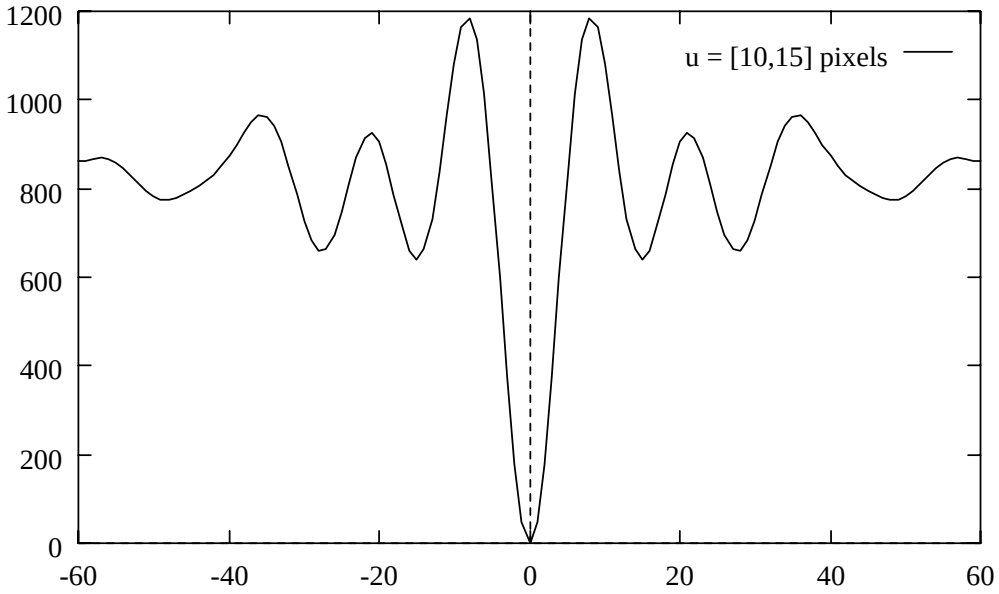

Figure 19: The energy function of the above image. The calculated B.O.A is 16 pixels.

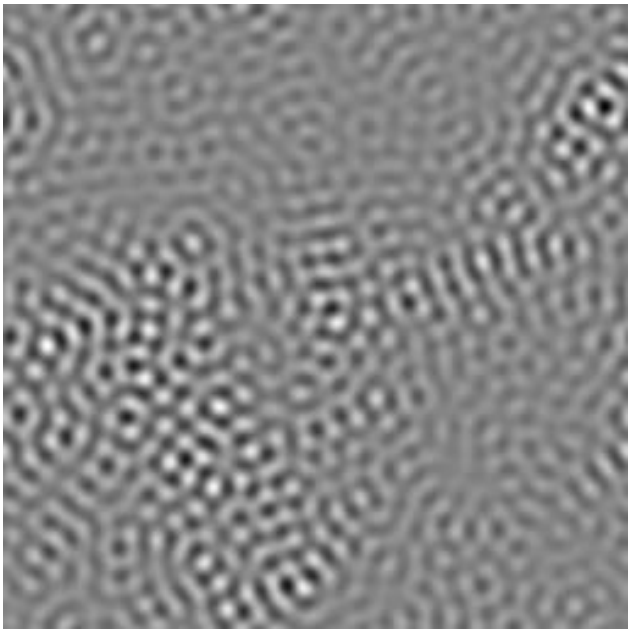

Figure 20: Band pass image with frequencies $15\Delta u \leq u \leq 20\Delta u$.

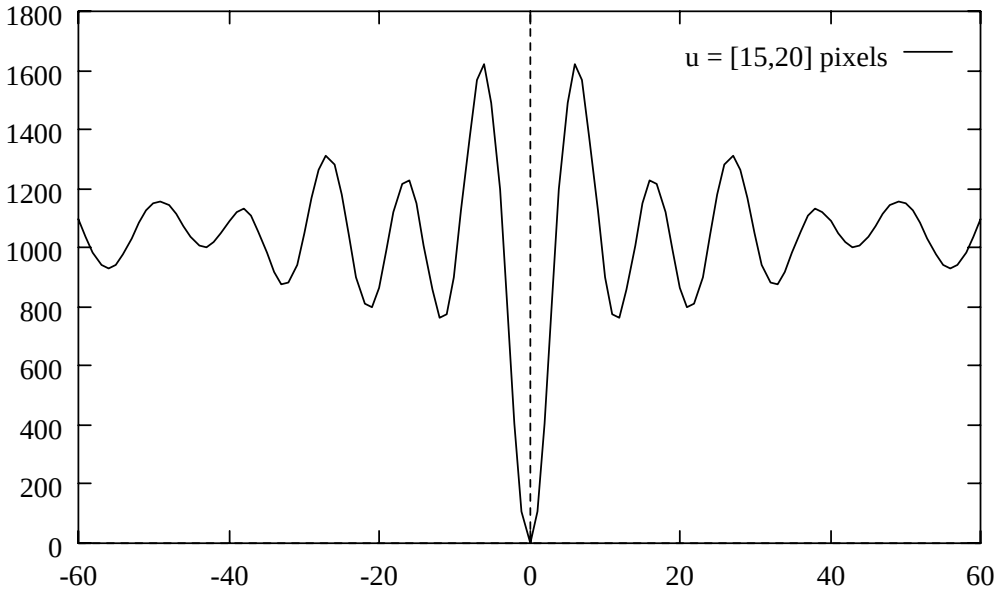

Figure 21: The energy function of the above image. The calculated B.O.A is 12 pixels.

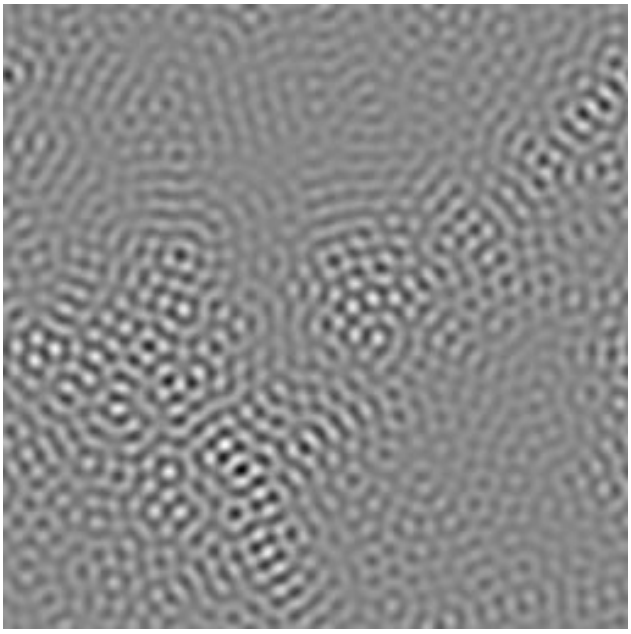

Figure 22: Band pass image with frequencies $20\Delta u \leq u \leq 25\Delta u$.

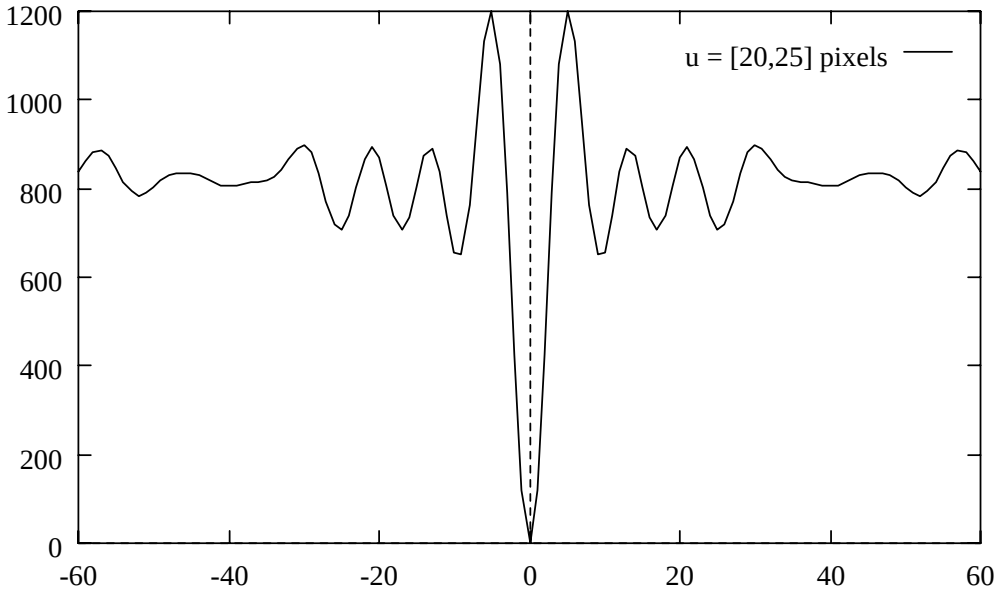

Figure 23: The energy function of the above image. The calculated B.O.A is 10 pixels.

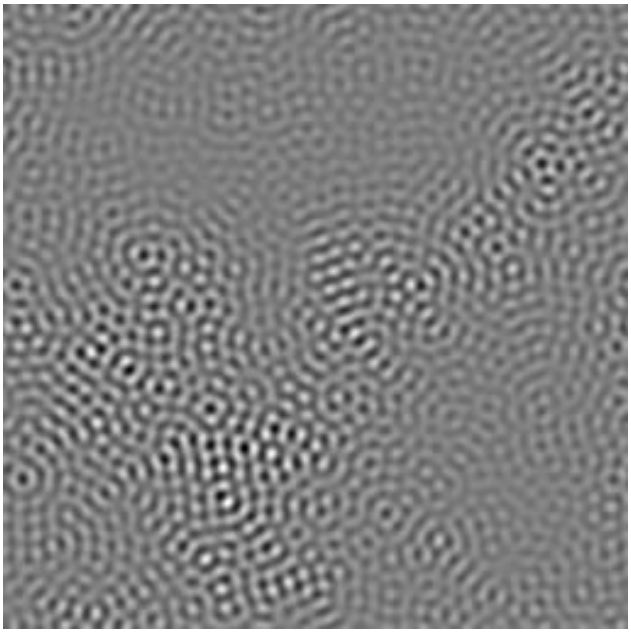

Figure 24: Band pass image with frequencies $25\Delta u \leq u \leq 30\Delta u$.

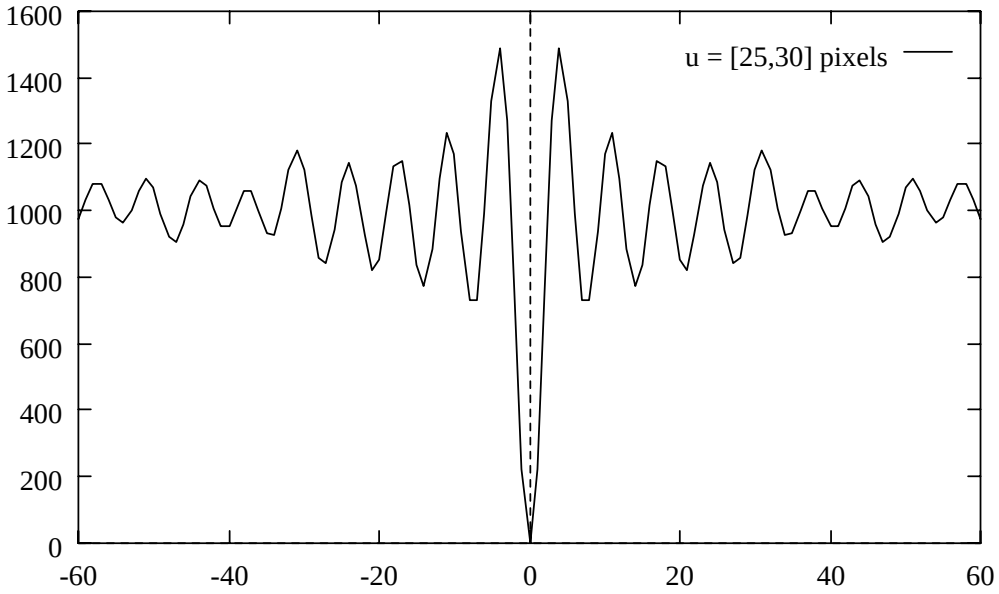

Figure 25: The energy function of the above image. The calculated B.O.A is 8 pixels.

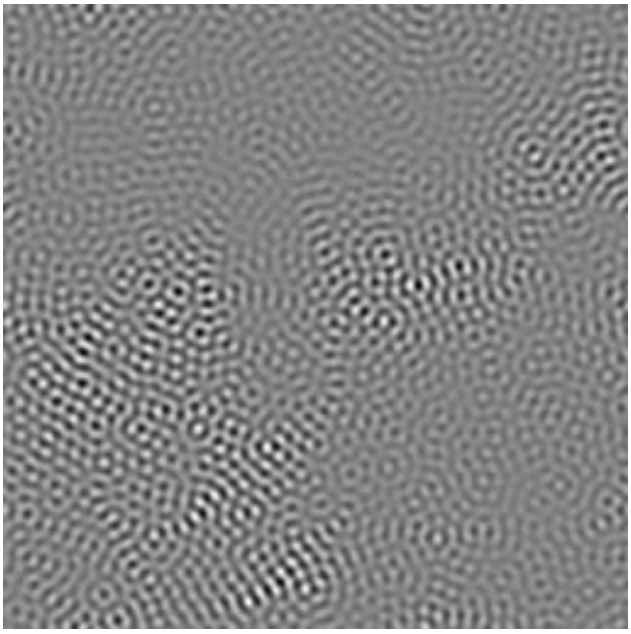

Figure 26: Band pass image with frequencies $30\Delta u \leq u \leq 35\Delta u$.

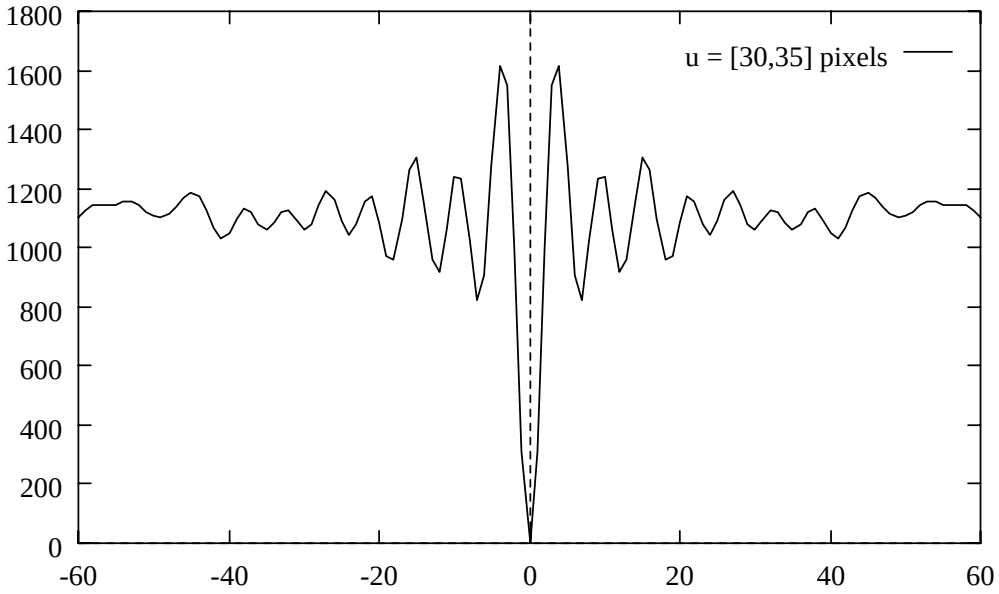

Figure 27: The energy function of the above image. The calculated B.O.A is 8 pixels.

6 Evaluate 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 $-p$: evaluating the angle θ is equivalent to estimate the speed p. However, since our function $I(x,t)$ have a finite support (or it is periodic in space and time) the continues 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:

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

where r is the maximim length 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 then 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 $-p'$ (grey line in Figure 10). This now allows us to take now 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 a 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 kind of sensors. To illustrate the problem assume that two images such as in Figure 2 were taken under different illumination condition. To simulate such a case the greyscales of one of the images were multimlied 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 **??**. It is demonstrated 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 condition causes "smearing" in the low frequencies in the spatio-temporal frequency space. Low pass filters which focuse 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 **??**.

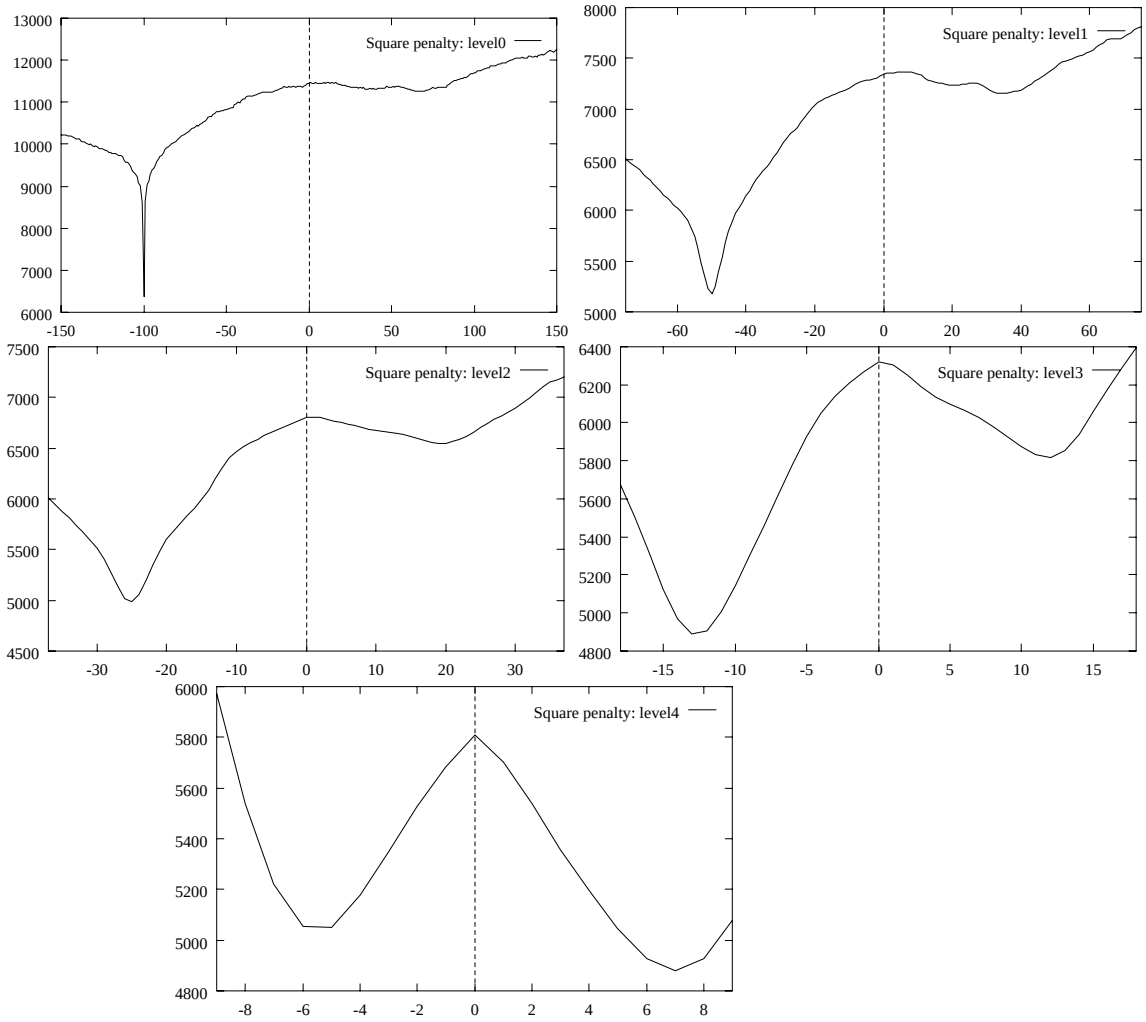

Figure 28: Different illumination condition: the energy function for 5 levels in the Gaussian Pyramid. The higher level fails to extract the global minimum.

Even harder case is where the greyscales of the two iamges are different due to diffenent kind of sensors taking these images. In this case the corresponding greyscales of the images can be totaly different even non-monotonic. The fact that the images show the same scene is expressed only in the hight frequencies of the images, therefore, a comparison can be made only at these frequency levels. However, using the high frequncies will drasticly reduce the stability of the convergence and the multiresolution scheme will not produce much more performance. An example can be seen in the following simple case. In Figure **??** two identical images were covered by a non-monotonic greyscales. Trying to register these two

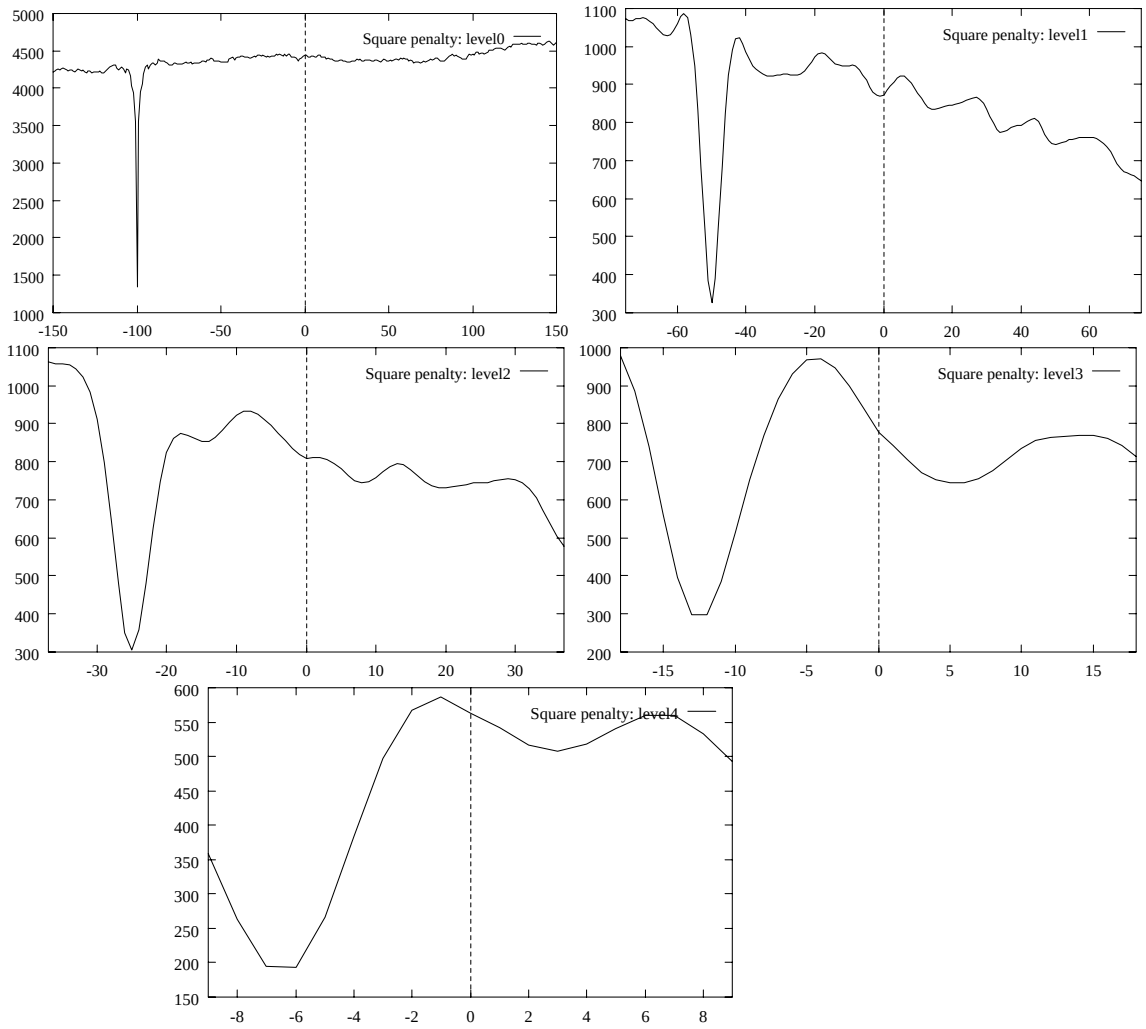

Figure 29: Different illumination condition: the energy function for 5 levels in the Laplacian Pyramid. The higher level enables to extract the global minimum.

images by minimizing $E(p)$ of the original images is hopeless. Trying to do so by the edge information of these images (high pass filter) is a better possibility. The Energy functions $E_k(p)$ for 4 levels in the Gaussian pyramid of the edge images are plotted in Figure **??**. 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 is not expanded a lot at the higher level of the pyramid. This results by a limited performance of the energy minimization scheme.

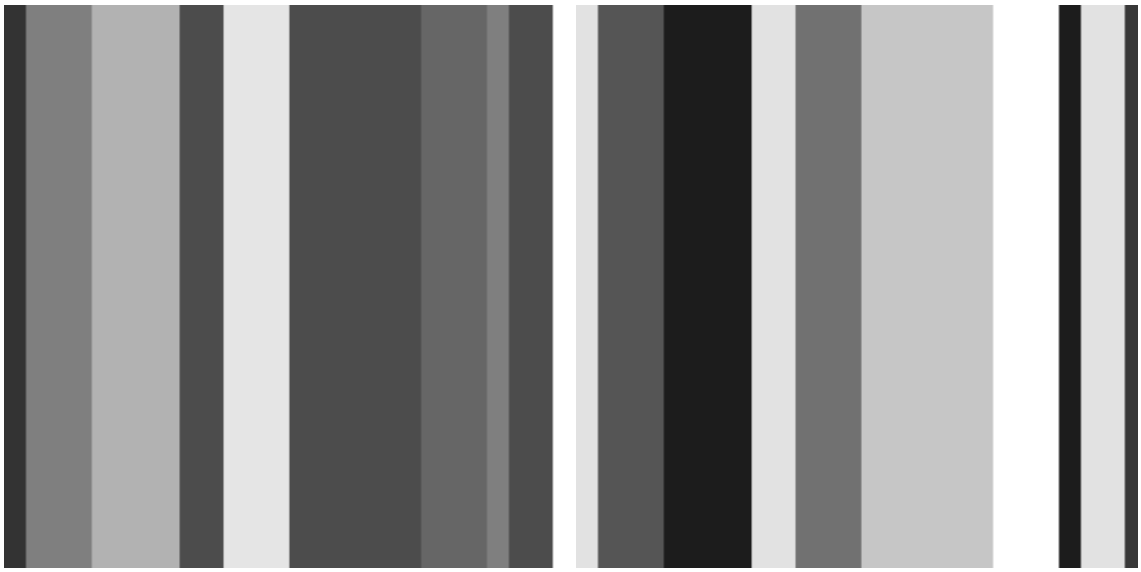

Figure 30: Two Images with Non-monotonic Grey Scales.

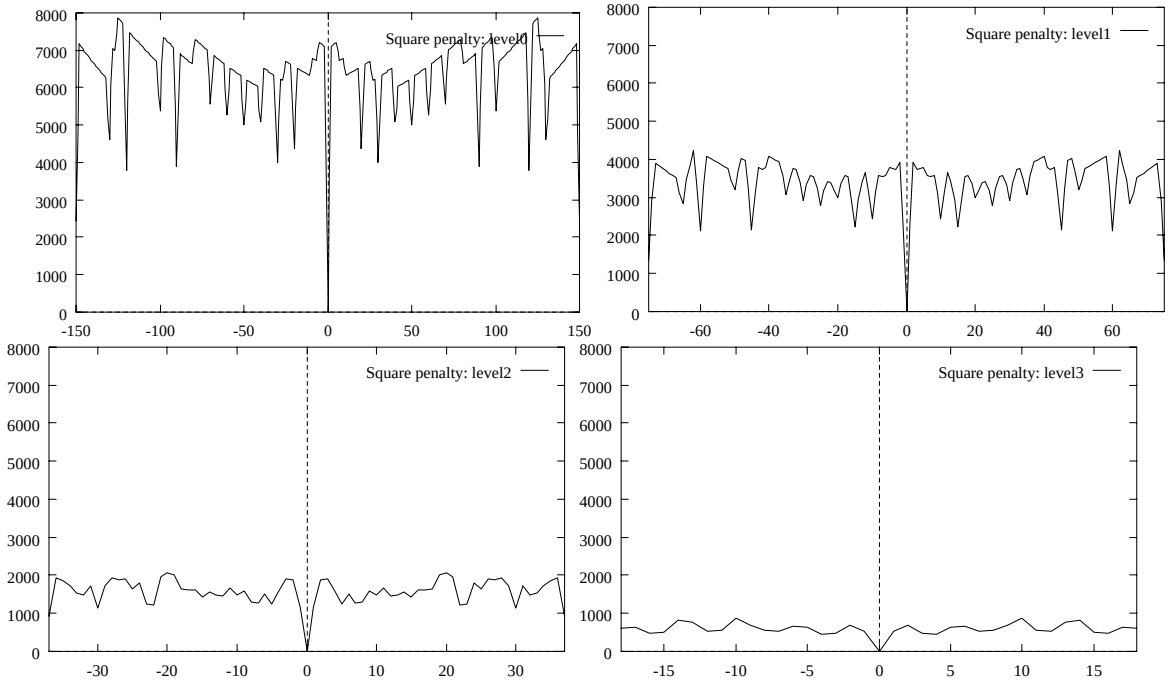

Figure 31: Edge images: the energy functions for 4 levels in the Gaussian pyramid. The B.O.A is quite narrow at all of the levels.

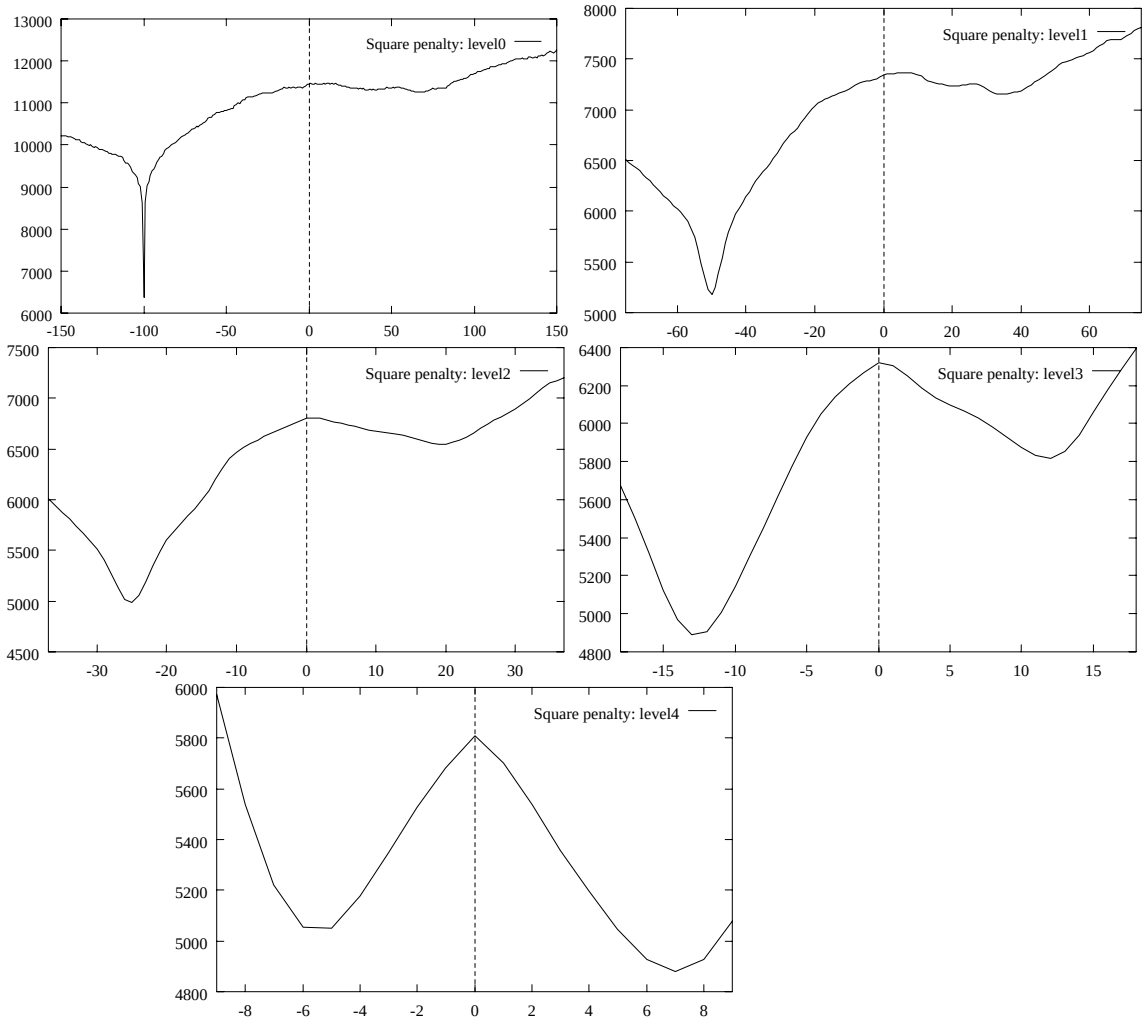

Figure 32: Different illumination condition: the energy function for 5 levels in the Gaussian Pyramid. The higher level fails to extract the global minimum.

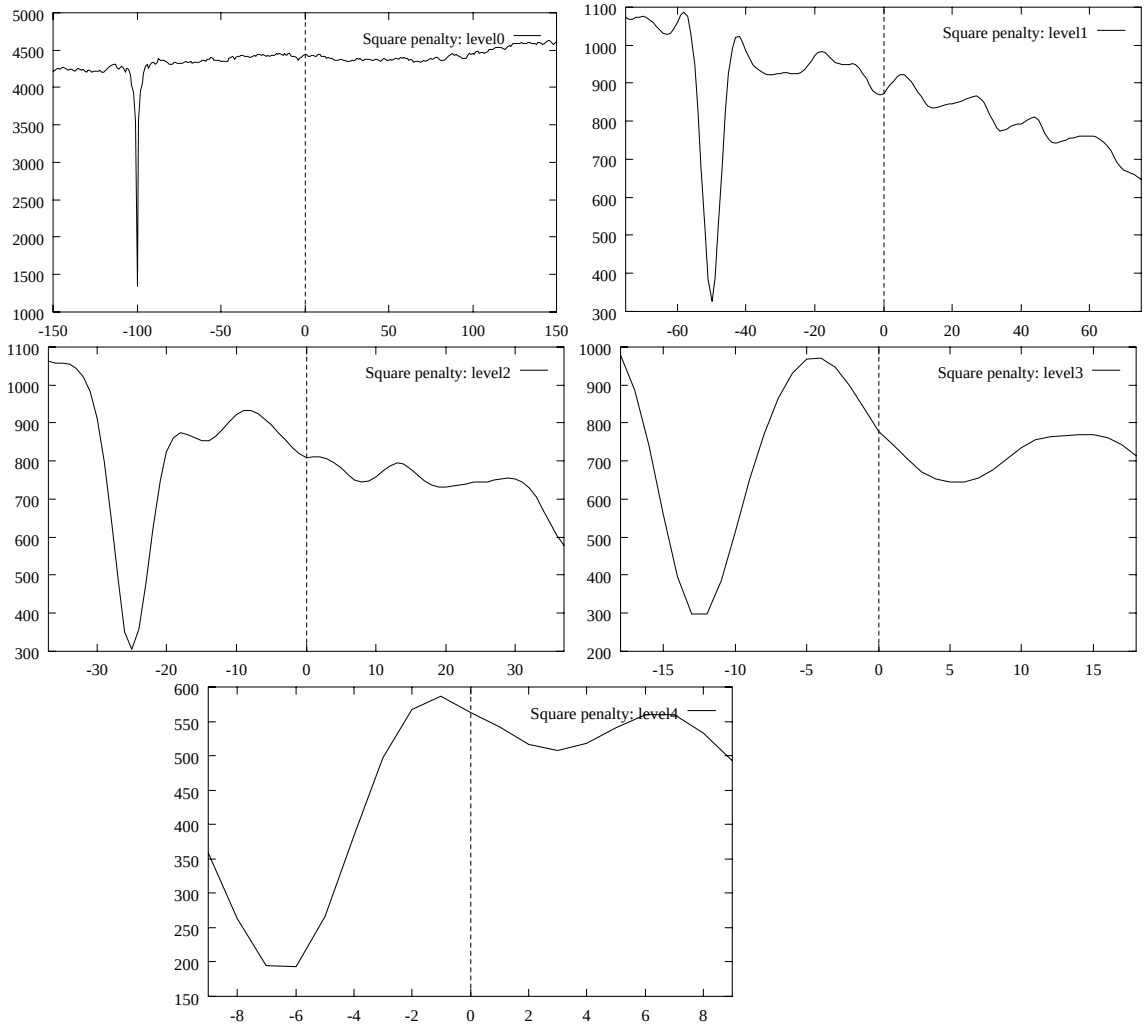

Figure 33: Different illumination condition: the energy function for 5 levels in the Laplacian Pyramid. The higher level enables to extract the global minimum.