Some Improvements to the Lucas-Kanade Optical Flow Detector in Motion Tracking

Zoltán Prohászka

Department of Control Engineering and Information Technology
Budapest University of Technology and Economics
Magyar tudósok krt. 2, H-1117 Budapest, Hungary
prohaszka@iit.bme.hu

Abstract: The properties of the Lucas-Kanade detector in motion tracking have been analysed considering the influence of typical signal forms, sampling methods and frequency domain filtering on the accuracy and computation time. Based on the investigation a simple convolution filter has been elaborated which is able to balance low and high frequency information in the images in a single step.

Keywords: motion-tracking, optical flow, hierarchical Lucas-Kanade detector, sampling effects, filtering

1 Introduction of the LK Detector

1.1 Brief Preview

The Lucas Kanade Optical flow detector is a wellknown tool in motion tracking. This detector is the basis of many image pairing algorithm. Image pairing is a first and critical step of nowadays 3D vision systems. These vision systems are intensively used in Robotics and vehicle control The main goal of this paper is to clarify these capabilities, so the developers get know what they can and can not expect at higher levels of image matching solutions. The Lucas-Kanade detector is a gradient based displacement detector. It does not give exact solution, only a relatively good solution. Theoretically, the accuracy is inversely proportional with the disparity (and thus with the displacement), this is why the original approach handles this problem by iterative calculation. 10..40 iterations are mentioned in [4]. In practice, the effect of intensity integration along a pixel causes the accuracy decrease in the case of very small displacements. Our investigation focused on the simulation of this problem. The hierarchical (or multi-resolution) approach is mentioned also in the first paper of Lucas and Kanade [1]. This approach changes the role of iterations, since by five processing stages we are 'iterating' five times by default. The question is, if it is possible to implement such a hierarchical LK detector, that one iteration per resolution stage is enough to get sub-pixel accurate final results.

The LK detector is commonly used to determine transformation between two images, if the displacement is not more than a few pixels. The hierarchical approach could extend its capability to handle greater, but still not so far displacements. On the other hand, the hierarchical approach has many occurring problems. It is useful to investigate the capabilities of the simple one stage detector prior to the development of a robust hierarchical LK detector. In this article we present the results of two interesting investigation, in conjunction with the requirements that are stood up by the results against a well-performing algorithm.

1.2 1D Case

The basic idea of the LK detector assumes that the displacement between two functions is infinitesimal, and the shapes of these functions are smooth (they contain frequencies corresponding to longer wavelengths than the grade of the displacement). Such a signal, if displaced by an infinitesimal ds , satisfies

$f_2(x) - f_1(x) = a \cdot ds$, where

$f_1(x) = f(x); \;\; f_2(x) = f(x + ds),$ and

a is the slope of the function $(a = df / ds)$ at the investigated location. If we measure f_1, f_2 as two different signals, we can compute an approximation to ds :

$$a = \frac{(f_1'(x) + f_2'(x))}{2},$$

$$\Delta f = f_2(x) - f_1(x),$$

$$ds \cdot a = \Delta f \tag{1}$$

It is intentional, that we did not expressed a exactly, since if we perform n measurements in n different locations, than ds must satisfy (1) for all $i \le n$. The simultaneous equations $ds \cdot a_i = \Delta f_i$ can not be satisfied, only their error can be minimised, which yields to a Least-Squares (LS) problem.

1.3 Formulas for the 2D Case

Extending the above deduction to the second dimension yields to the following formulas:

$\Delta x_i = (df_1/dx + df_2/dx)/2$ at $x_i; y_i$

$\Delta x_i = (df_1/dx + df_2/dx)/2$ at $x_i; y_i$

$\Delta f_i = f_2 - f_1$ at $x_i; y_i$

(this is known also as the images' derivative respect to time, Δt_i)

$$[\Delta p_x \quad \Delta p_y] \cdot \begin{bmatrix} \Delta x_i \\ \Delta y_i \end{bmatrix} := \Delta f_i \cdot$$

The computation of the derivatives is done by the following masks:

$$\overline{\overline{M_x^{f1}}} = \frac{1}{4}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}; \qquad \overline{\overline{M_x^{f2}}} = \frac{1}{4}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix};$$

$$\overline{\overline{M_y^{f1}}} = \frac{1}{4}\begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix}; \qquad \overline{\overline{M_y^{f2}}} = \frac{1}{4}\begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix};$$

$$\overline{\overline{M_f^{f1}}} = \frac{1}{4}\begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix}; \qquad \overline{\overline{M_f^{f2}}} = \frac{1}{4}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix};$$

These formulas give proper results for images having a pure, small shift, and having no frequencies below the wavelength of few pixels.

However, low frequency images with a transformation other than a shift operation contain different displacement at each location. The common method to handle this is to divide the screen into sensor windows, assuming that the displacement within each window can be considered as constant. The window size is a parameter (the only one) of the 2D LK detector, typically 8..32. [3].

1.4 Short Description of the Hierarchical Approach

The hierarchical approach takes over the one-pixel displacement limit by drastically enlarging pixel size. This is done recursively by integrating 2*2 pixel windows into one pixel of the new image. On the other hand this method drastically reduces image details. Therefore, after determining the displacement on the lowest resolution version of the images, both original images are corrected by the detected optical flow, and fine details of the flow are determined by using higher and higher resolution images.

1.5 Affine Extension

Another problem occurs, when the image in the window of interest satisfies the frequency and displacement level, but contains a high amount of rotation, or other affine transformation [1],[4], see Fig. 1.

Figure 1
Two superimposed images showing rotation around 15 degrees

If we modify the elementary detector to look for an affine transformation, not for a simple shift, than this problem can be solved:

Let the searched transformation be:

$$\begin{bmatrix} \Delta p_x \\ \Delta p_y \\ 0 \end{bmatrix} = \begin{bmatrix} & \overline{\overline{A_{2*3}}} & \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} p_x^i \\ p_y^i \\ 1 \end{bmatrix}.$$

At each 2*2 measure window, we have the following (measured) relation:

$$\begin{bmatrix} \Delta p_x & \Delta p_y \end{bmatrix} \cdot \begin{bmatrix} \Delta x_i \\ \Delta y_i \end{bmatrix} := \Delta f_i;$$

substituting

$$\begin{bmatrix} a_{11} p_x^i + a_{12} p_y^i + a_{13} \\ a_{21} p_x^i + a_{22} p_y^i + a_{23} \end{bmatrix} \text{ into } \overline{\Delta p},$$

we get a linear relation to the elements of $\overline{\overline{A}}$:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{21} & a_{22} & a_{23} \end{bmatrix} \cdot \overline{v_i} := \Delta f_i, \text{ where}$$

$$\overline{v_i} = \begin{bmatrix} p_x^i \Delta x_i & p_y^i \Delta x_i & \Delta x_i & p_x^i \Delta y_i & p_y^i \Delta y_i & \Delta y_i \end{bmatrix}^T$$

Solving the resulting LS problem can get the best solution for $\overline{\overline{A}}$.

If the main goal of the usage of the LK detector is to determine an affine transformation between two images, than the elementary sensor window can be enlarged to incorporate the whole picture.

2 Capabilities of the LK Detector

2.1 Typical Images

Applying a simple, downloadable LK detector to real images, it can be realised that the displacement could be greater than one pixel in many cases. Fig. 2 shows a typical case for images in an enlarged sensor window:

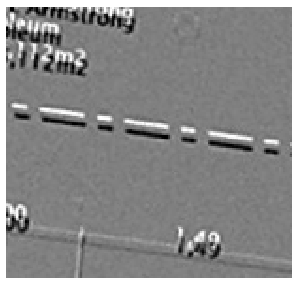

Figure 2
Typical input: The difference of two shifted images with 50% grey bias is shown

We investigated, what is the best method to handle displacements of 5..10 pixels. First, a series of basic experiments was carried out, to clearly discover the capabilities of the detector. The one-dimensional case was chosen for these tests.

2.2 One Dimensional Tests

2.2.1 Used Signal Forms

The following signals were used to test the basic 1D detector, see Fig. 3:

- rectangular impulse
- unit step
- equilateral triangle
- unit ramp
- continuous sine wave
- vertically shifted (co)sine wave of one period (Hanning window)

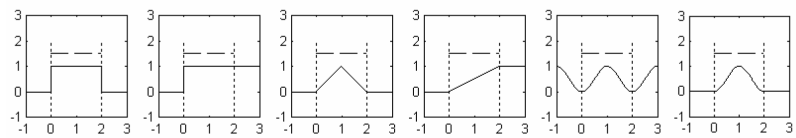

Figure 3

The signals (the signal width parameter was chosen to 2 for this figures)

2.3 Handling the Sampling Effect

Real, advanced image recording devices integrate image intensity over a pixel, which has a strong effect on details if their sizes are around or less than one pixel. Usually, the LK detector is expected to produce results with sub-pixel accuracy. Therefore, it was considered to be useful to simulate the sampling phenomenon. To carry out the integral of the algorithmically given function $f(x)$ between x_1 and x_2 is easy, if not only $f(x)$ but the primitive function of it is given (implemented) algorithmically. One can recognise that it is not necessary to be able to evaluate $f(x)$ for these tests if one can evaluate the primitive function. Therefore, we implemented the above mentioned waveforms through their integrals.

2.4 Results

Each of the test signals was shifted by a series of displacement. Then, the original and the shifted signal were shifted by a random value before applying the sampling effect. After this, the 1D LK detector was applied and the calculated displacement wes compared to the original one. The measurements were collected into a table, which shows the ratio of the detected and the real displacement for the used test signals with various width and displacement levels. Since the table would have more thousand entries, hence only a compressed representation is shown, see Table 1 and 2 containing: horizontal: width; vertical: signal type; value: the real displacement (pixels) interval, for which the accuracy is within the 80-125% boundary.

Table 1

Results: Spatial domain signals

Signal	Signal parameter (width in pixels)					
	0.5	1	1.5	2	3	4
Impulse	0.5-1.1	0.8-1.2	0.6-1.6	0.3-2	0-1.9	0-1.8
Unit step	0-1.9	(this signal do not have width)				
Triangle	0.8-1.2	0.6-1.2	0.4-1.5	0.7-1.5	0-1.9	0-2.4
Unit ramp	0.3-1.9	0-2.2	0-2.6	0-3.2	0-4.4	0-5.5

Table 2
Results: frequency domain signals

Signal	Signal parameter (wavelength in pixels)					
	2	3	4	6	8	10
Sine wave	-	0.8-1.1	0.5-1.2	0-1.7	0-2.2	0-2.5
Hanning	0.5-1.4	0.5-1.7	0-2.1	0-3.1	0-4.2	0-5.1

If we look after results where the absolute error is less than 0.5 pixel, we get similar results as by the relative error limit.

2.5 Consequences

2.5.1 Spatial domain

Good result is if the shift is less or around 1 pixel for lower frequency signals.

2.5.2 Frequency domain

Good result is for shifts less than $\pi/2$ for pure sinusoidal signals, and less than π for the Hanning signal. The detected translation is within the 80%-125% interval in the case of wavelengths greater than 4 pixels.

2.6 Conclusion

The best way to handle displacements seen in the Fig. 2 is to use only low frequency spatial information, which can be done by a simple low pass 2D convolution with, or without an additional lower resolution resampling. The undersampling is part of the standard hierarchical processing, but the additionally used filter should be designed by the consideration of the described experiment.

2.7 Problem of Low Frequency Noise

We used the detector to match details of two previously alligned images. Both of them contained the same scenario, but different shades were superimposed on them. However, this type of intensity error between the images causes the detector to fail. We found that high-pass filtering the image helps to solve this problem, since the shading has effect on few processing stages only. We have choosen the cut-off frequency of the high pass filter to be the double of the low-pass filter's cut off.

2.8 Requirements Against a Filter for each Hierarchical Stage

As we can see, at first glance, a filter that transmits only the octave of $4.5 < \lambda < 9$ is favourable. It is useful to set up another requirement, which takes the hierarchical approach into account: If there is a component having the frequency f, and it is subjected to go through the filters of each hierarchical stage (placed one octave apart), then the sum of amplitudes of the filtered signals (of the same frequency) should be the same for any frequency. This property assures that every image frequency contributes to the final result by the same grade.

2.8.1 Formulas

If we denote the filters transfer characteristic as $W(f)$, then the above mentioned requirements can be expressed in compact form:

$$\forall f : \sum_{i=-\inf}^{\inf} W(f \cdot 2^i) \ := c,$$

where c is a constant, around 2, if

$$\max W(f) = 1.$$

Let us choose three wavelengths, which will be analysed in the terms of the above expression, $\lambda_1 = 6$, $\lambda_2 = 5$, $\lambda_3 = 4$. Let us assume that the current hierarchical stage was preceded and will be followed by other stages, which will be indexed relatively to this stage. The index of the current stage is 0. The contribution of this wave to several processing stages will be:

$$W_1 = W(f * 0.5), \quad W_0 = W(f), \quad W_{-1} = W(f * 2), ...$$

Based on the above requirements and concerning simplicity and computational requirements, the following 2D filter mask has been elaborated and experimentally tested:

$$\begin{bmatrix} -1 & -5 & -11 & -14 & -11 & -5 & -1 \\ -5 & -25 & -55 & -70 & -55 & -25 & -5 \\ -11 & -55 & 23 & 134 & 23 & -55 & -11 \\ -14 & -70 & 134 & 380 & 134 & -70 & -14 \\ -11 & -55 & 23 & 134 & 23 & -55 & -11 \\ -5 & -25 & -55 & -70 & -55 & -25 & -5 \\ -1 & -5 & -11 & -14 & -11 & -5 & -1 \end{bmatrix} \cdot 1/2304$$

Which has an equivalent 1D convolution as:

$$\begin{bmatrix} -1 & -5 & 1 & 10 & 1 & -5 & -1 \end{bmatrix} / 48$$

The transfer values of this filter for the tested wavelengths at different stages are summarised in Table 3:

Table 3
Transfer characteristics

Stage #	3	2	1	0	-1	Total contrib.
λ'(pixels)	48	24	12	6	3	-
W(1/λ')	0.02	0.09	0.33	0.87	0.58	1.89
λ'(pixels)	40	20	10	5	2.5	-
W(1/λ')	0.03	0.13	0.44	0.98	0.23	1.81
λ'(pixels)	32	16	8	4	2	-
W(1/λ')	0.05	0.19	0.62	0.97	0.0002	1.83

3 Computational Requirements

3.1 Running Time of the Affine Extension

It is necessary to compare the computational time of the standard and the affine LK detector [2]:

3.1.1 Standard LK Detector

The computational time of the standard LK detector on two images containing m pixels, with a n*n sensor window can be expressed as:

T=number_of_pixels*(calc.of_derivatives+accumulation_for_the_LS_solution)+ number_of_windows* 2D_LS_solution=

m*((9 add +4 far_read)+(4 add+4 mul+2 add+2 mul))+m/(n*n)*T_2D_LS=

m*(15 add+ 4 far_read +6 mul)+m/(n*n)*T_2D_LS=

m*25 op +m/(n*n)*(2 mul+1 add+ 1 div+ 4 mul+ 2 add)=

m*(25 op +1/(n*n)*(9 op+ 1 div))

The division usually takes much longer than other operations denoted as ‘op’.

3.1.2 Affine LK Detector

The computational time of the extended affine LK detector can be expressed as:

T=number_of_pixels*(calc.of_derivatives+accumulation_for_the_LS_solution)+1 *T_6D_LS_solution=

m*(9 add+4 far_read + 4 mul+36 mul+36 add+6 mul+6 add)+T_6D_LS=

m*(51 add+ 4 far_read +42 mul)+ 6D LS=m*97 op+T_6D_LS

If we count each type of operation to take the same time, than the affine algorithm is four times slower than the standard one. In the case of big images, this can be reduced.

Conclusions and Further Research

In this paper we shared the results of basic experiments on the Lucas-Kanade detector in motion tracking regarding signal form in the image, sampling technique, and frequency domain properties. Based on the investigation a simple convolution filter has been elaborated which is able to balance low and high frequency information in the images in a single step. We found these results to be useful in the implementation and improvement of such a detector.

Our final goal is obtain a robust image pairing algorithm which can handle higher amount of disparity than observed on video sequences. The implementation of a robust hierarchical detector is in progress. Several extensions will be tested to provide significance information for higher processing stages.

Acknowledgement

The research was supported by the Hungarian National Research Program under grant No. OTKA T 042634.

References

[1] B. D. Lucas, T. Kanade: An Interactive Image Registration Technique with an Application to Stereo Vision, International Joint Conference on Artificial Intelligence, pp. 674-679, 1981

[2] S. Baker and I. Matthews, Lucas-Kanade 20 Years On: A Unifying Framework, International Journal of Computer Vision, Vol. 56, No. 3, March, 2004, pp. 221-255

[3] C. Tomasi, T. Kanade: Detection and Tracking of Point Features, Carnegie Mellon University Technical Report CMU-CS-91-132, 1991

[4] J. Shi and C. Tomasi: Good Features to Track, IEEE Conference on Computer Vision and Pattern Recognition, pp. 593-600, 1994

[5] R. Hartley, A. Zissermann: Multiple View Geometry in Computer Vision, Cambridge University Press, 2003