Optical Flow Measurement using Lucas kanade Method

Dhara Patel,
ME-CSE student of Computer science and Engineering,
S.P.B. Patel Engineering College, Saffrony, Mehsana,
India.

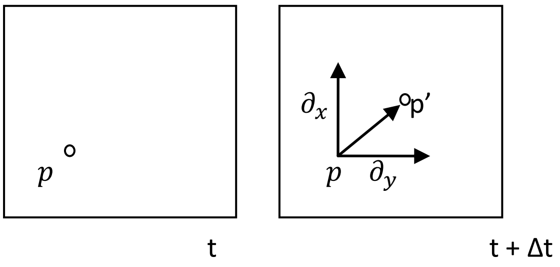
Saurabh Upadhyay,
Asso.Prof. of Computer science and Engineering
Department,
S.P.B. Patel Engineering College, Saffrony, Mehsana,
India.

ABSTRACT

Motion estimation is demanding field among researchers to compute independent estimation of motion at each pixel in most of general. Motion estimation generally known as optical or optic flow. In this paper, overview of some basic concepts of motion estimation, optical flow and Lucas kanade method has been provided by us. Lucas kanade method is one of the methods for optical flow measurement. It is a differential method for optical flow estimation.

1. INTRODUCTION

Motion is very important part in image sequences. Motion estimation is the process of determining motion vectors that describe the transformation from one 2-D image to another [1]. According to Jaling wu [1]-motion estimation mean the estimation of velocity of image structures from one frame to another frame in time sequence of 2-D image.



$$\partial = [\partial_x, \partial_y]^T$$

Fig.1: 2-D image of the pixel location.

Fig.1 shows the 2-D displacement of the pixel located at point p from frame at time t to frame at time $t + \Delta t$**.** The motion field can almost always be unambiguously related to translational and rotational velocities of rigid surfaces [1]. These are the following equations for translation and rotation.

Translation-$x' = x + t,\ x' = [I\ t]\bar{x}$
Here *I* is identity matrix.

Rotation + Translation:

$$x' = Rx + t$$

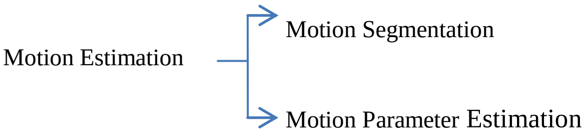
Or

$$x' = [R, t]\,\bar{x}$$

Where

$$R = \begin{bmatrix} \cos\emptyset & -\sin\emptyset \\ \sin\emptyset & \cos\emptyset \end{bmatrix}$$

The motion estimation is mainly classified into two categories. Ja-ling wu suggested,



Motion segmentation means to identify the moving object boundaries in frame. And number of parameters included in motion estimation like rigid bodies, translation movement, rotation and illumination etc.

Motion estimation techniques are also used in many other applications: –Computer vision, target tracking, industrial monitoring. The motion vectors may relate to the image or its parts such as blocks, pixels etc. Chiara Ercole [2] suggested that 2D motion results from the projection of moving 3D objects on the image plane. The 2D projection is called apparent motion or optical flow. The projected motion recovered from intensity variation information of a frame sequence. Fig.2 [4] defines various motion fields. Generally there are three types of motion such as: forward, horizontal and rotation.

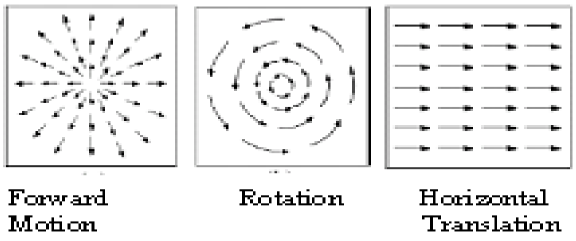


Fig.2: various motion fields.

2. OPTICAL FLOW

Optical flow or optic flow is the pattern of apparent motion of objects, edges and surface in a visual scene caused by the relative motion between an observer (an eye or a camera) and the scene [3].

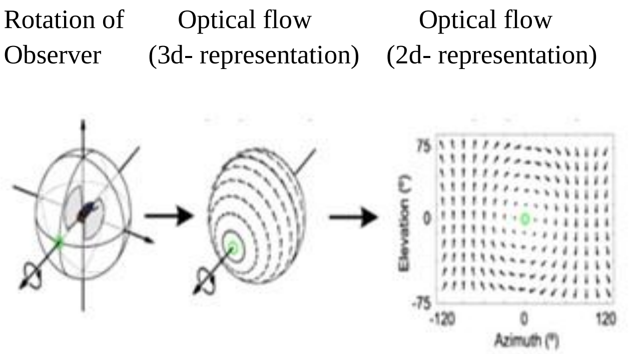

Rotation of Observer
Optical flow (3d- representation)
Optical flow (2d- representation)
75
0
-75
-120
0
120
Elevation (°)
Azimuth (°)

Fig.3: The optical flow experienced by a rotating observer.

Fig.3 [3] defines the direction and magnitude of optic flow at each location and also defines the length of each arrow. Fig.4 [4] defines the relation between Motion Field and Optical Flow Field.

- Motion Field = Real world 3D motion
- Optical Flow Field = Projection of the motion field onto the 2d image.

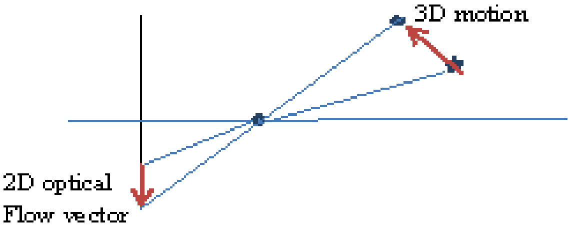


Fig.4 : The relation between motion field & optical flow field.

J.L.Barron and N.A.Thacker [5] suggested that 2D Motion Field can be defined as -2D velocities for all visible points. And Optical Flow Field can be defined as- Estimate of the 2D motion field. Optical flow is an approximation of the local image motion based upon local derivatives in a given sequence of images. Optical flow techniques such as motion detection, object segmentation, time-to-collision and expansion calculations, motion encoding, and stereo disparity measurement utilize this motion of the objects' surfaces and edges.

The 2D image sequences used here are formed under perspective of projection via the motion of a camera and scene objects. The 3D volume sequences used here were formed under orthographic projection for a moving object. The optical flow is tightly linked to the recovery of the 2D projected motion derived from acquisition and projection on the image plane of 3D objects motion in the real world [5]. All the general considerations made on 3D motion projection are still valid dealing with optical flow.

Optical flow constraints (images) according to [6] would be.

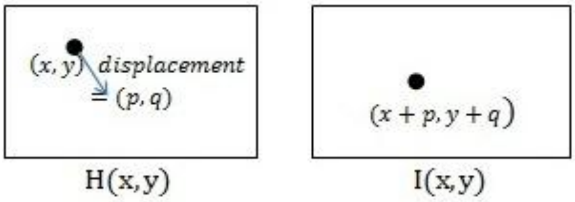


Fig.5: 2D motion of pixel.

The calculations of optical flow are as follows:

$\mathrm{O}= \mathrm{I}(\mathrm{x}+\mathrm{p},\ \mathrm{y}+\mathrm{q})-\ \mathrm{H}(\mathrm{x},\mathrm{y})$

$\approx \mathrm{I}(\mathrm{x},\mathrm{y})\ + I_x p + I_y q -\ \mathrm{H}(\mathrm{x},\mathrm{y})$

$\approx (\mathrm{I}(\mathrm{x},\mathrm{y})-\ \mathrm{H}(\mathrm{x},\mathrm{y})) + I_x p + I_y q$

$\approx I_t + I_x p + I_y q$

$\approx\ I_t +\ \nabla \mathrm{I}\ \cdot [p, q]$

In the limit as u and v go to zero, this becomes exact

$$O = I_t +\ \nabla \mathrm{I}\ \cdot [\frac{\partial x}{\partial t}\frac{\partial y}{\partial t}]$$

The 2D and 3D Motion estimation of optical flow according to J.L.Barron and N.A.Thacker [5].

2D motion equation - Assume $I(x, y, t)$ is the center pixel in a $n \times n$ neighbourhood and moves by $\delta_x,\ \delta_y$ in time δ_t to $(x+\delta_x, \mathrm{y}+\delta_y, \mathrm{t}+\delta_t)$. Since $I\ (x, \mathrm{y}, \mathrm{t})$and $I\ (x+\delta_x, \mathrm{y}+\delta_y, \mathrm{t}+\delta_t)$are the images of the same point (and therefore the same) we have [5]:

$$I\ (x, \mathrm{y}, \mathrm{t}) = I\ (x+\delta_x, \mathrm{y}+\delta_y, \mathrm{t}+\delta_t). \qquad (1)$$

This assumption for the 2D Motion Equation. The small local translations provided δ_x,δ_y,δ_t are not too bigger. We can perform a 1st order Taylor series expansion about $I\ (x, \mathrm{y}, \mathrm{t})$ in equation (1) to obtain [5]:

$$I\ (x+\delta_x, \mathrm{y}+\delta_y, \mathrm{t}+\delta_t)\ =\ I\ (x, \mathrm{y}, \mathrm{t}) + \frac{\partial I}{\partial X}\delta_x + \frac{\partial I}{\partial y}\delta_y + \frac{\partial I}{\partial t}\delta_t +\ \mathrm{H.O.T.}, \qquad (2)$$

Where H.O.T. is the Higher Order Terms, which we assume are small term can safely be ignored. Using the above two equations we obtain:

$\frac{\partial I}{\partial X}\delta_x + \frac{\partial I}{\partial y}\delta_y + \frac{\partial I}{\partial t}\delta_t = 0$ Or

$\frac{\partial I}{\partial X}\frac{\delta x}{\delta t} + \frac{\partial I}{\partial y}\frac{\delta y}{\delta t} + \frac{\partial I}{\partial t}\frac{\delta t}{\delta t} = 0$ And finally we get

$$\frac{\partial I}{\partial X}v_x + \frac{\partial I}{\partial y}v_y + \frac{\partial I}{\partial t} = 0$$

Here $v_x = \frac{\delta x}{\delta t}$ and $v_y = \frac{\delta y}{\delta t}$ are the x and y components of image velocity or optical flow and$\frac{\partial I}{\partial X}$, $\frac{\partial I}{\partial y}$ and $\frac{\partial I}{\partial t}$are image intensity derivatives at $x, \mathrm{y}, \mathrm{t}$. We normally write these partial derivatives as:

$$I_x = \frac{\partial I}{\partial X}, I_y = \frac{\partial I}{\partial y} \text{ and } I_t = \frac{\partial I}{\partial t}$$

(I_x, I_y, I_t) are intensity derivatives. This equation can be rewritten more compactly as:

$$(I_x, I_y) \bullet (V_x, V_y) = -I_t$$
$$\nabla I \cdot \tilde{v} = -I_t$$

Where $\nabla I = (I_x, I_y)$is the spatial intensity gradient and $\tilde{v}$=(V_x ,V_y)is the image velocity or optical flow at pixel (x, y) at time t. $\nabla \mathrm{I} \cdot \tilde{v} = -\mathrm{It}$ is called 2D motion.

3D motion equation- The 3D motion equation is a simple extension of the 2D motion equation. In 3D $n \times n \times n$ block at (x, y, z) at time t moving by $(\delta_x, \delta_y, \delta_z)$ to $(x+\delta_x, \mathrm{y}+\delta_y, \mathrm{z}+\delta_z, \mathrm{t}+\delta_t)$ over timeδ_t. Since $I\ (x, \mathrm{y}, \mathrm{z}, \mathrm{t})\ =\ I\ (x+\delta_x, \mathrm{y}+\delta_y, \mathrm{z}+\delta_z, \mathrm{t}+\delta_t)$ we can perform a 1st order Taylor series expansion and obtain [5]:

$$I_x V_x +\ I_y V_y + I_z V_z + I_t = 0$$

Where $\widetilde{v} = (V_x, V_y, V_z) = (\frac{\delta x}{\delta t}, \frac{\delta y}{\delta t}, \frac{\delta z}{\delta t})$ is the 3D volume velocity or 3D optical flow and Ix, Iy, Iz and It are the 3D spatio-temporal derivatives. This Equation can be written more compactly as:

$$\nabla I \cdot \tilde{v} = -\mathrm{It}$$

Where $\nabla I\ =\ (I_x, I_y, I_z)$ is the 3D spatial intensity gradient and$(V_x, V_y, V_z)$is the 3D velocity.

3. OPTICAL FLOW BASED MOSTION ESTIMATION

There are several methods for motion estimation [2]:

- Feature matching methods;
- Pel-recursive methods;
- Deterministic model based methods;
- Stochastic model based methods;
- Optical flow based methods;

There are several techniques which have been developed to approach the computation of optical flow. Such as [2]

- Gradient based approach
- Correlation-based approach
- Spatio-temporal energy based approach
- Phase-based approach

A. Gradient based approach

In this approach first method being proposed is the Horn and Schunck. This method initially defines moving object and sensor. The second method is the Lucas and Kanade. The iteration is not done in a strict pixel by pixel way, but considering a small neighbourhood of the pixel [2]. So, a minimization rule is used, corresponding to a least square fit of the brightness invariance constraint.

B. Correlation-based approach

This method is very close to block matching. The frame is here divided into fixed-size blocks and also this frame matching by a searching window. The displacement to the block that shows the highest likelihood or the lowest dissimilarity is taken as measure of the translational motion vector to our reference block [2]. The first method for this approach is the anandan's method. This computation starts at the highest level (at the lowest resolution) and a full search is performed by this method. The highest level is the one at maximum resolution. The passage from a higher level, L+1, to a lower one, L is done by interpolation and then subtraction from the upper level, L+1.What is the error image at level L. So the image at level L can be recovered by adding the error at level L to the interpolated estimate at level L [2].

The Singh's method, divides the information in two categories: the conservation information and the neighborhood information. Conservation information-estimate propagates in the neighboring area and then it iteratively updates the motion vector. Neighborhood information - The neighbourhood information are collected from areas surrounding our interest pixel which have sufficient intensity variations.

C. Spatio-temporal energy based approach

In the spatial temporal frequencies define the velocity of translational moving object. The signal corresponding to the movement is composed by several spatial frequencies. Motion can be evaluated by using Spatio-temporally oriented filters.

D. Phase-based approach

Such as differential techniques show better accuracy but are less efficient in implementation and lack a single useful confidence measure. The 3-D structure tensor technique yields the best results with respect to systematic errors and noise sensitivity.

According to [8] optical flow based techniques correctly estimate normal flow. If the temporal sampling theorem can be assured to be fulfilled, optical flow based techniques are generally the better choice. In other cases, when large displacements of small structures are expected, correlation-based approaches usually perform better.

In the section 4, explains one technic of optical flow which is Lucas kanade technic and its calculation for the video.

4. LUCAS KANADE METHOD FOR OPTICAL FLOW MEASUREMENT

The Lucas–Kanade method is a widely used in differential method for optical flow estimation and computer vision [9]. This method solves the basic optical flow equations for all the pixels in that neighbourhood, by the least squares criterion. it is a purely local method because it cannot provide flow information in the interior of uniform regions of the image. Here assumes that the flow is essentially constant in a local neighbourhood of the pixel under consideration [9].

The advantages and disadvantages of Lucas kanade method is: Advantages –this method is easy compare another method, very fast calculation and accurate time derivatives. Disadvantage - errors on boundaries of moving object [10].

Thus the optical flow equation can be assumed to hold for all pixels within a window centered at p. namely, the velocity vector (V_x, V_y) for local image must satisfy. The Lucas-Kanade method assumes that the displacement of the image contents between two nearby frames are small and approximately constant within a neighborhood of the point p [9].

$$
\begin{gathered}
I_x(p_1)V_x + I_y(p_1)V_y = -I_t(p_1) \\
I_x(p_2)V_x + I_y(p_2)V_y = -I_t(p_2) \\
. \\
. \\
. \\
I_x(p_n)V_x + I_y(p_n)V_y = -I_t(p_n)
\end{gathered}
$$

Where P1,P2…Pn are pixels inside the window, $I_x(p_n), I_y(p_n), I_t(p_n)$are the partial derivatives of the image I with respect to position x, y and time t, evaluated at the point Pn and at the current time. The equations can be written in matrix form:

$$Av = b, \qquad where$$

$$
\text{A=}\begin{bmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ . & . \\ . & . \\ I_x(p_n) & I_y(p_n) \end{bmatrix}\text{,v=}\begin{bmatrix} V_x \\ V_y \end{bmatrix}\text{, b=}\begin{bmatrix} -I_t(p_1) \\ -I_t(p_2) \\ . \\ . \\ -I_x(p_n) \end{bmatrix}
$$

Using the least squares principle determined the solution of Lucas kanade [9].

Now$A^T AV = A^T b$, we can write this equation in another form so we get:

$$V = (A^T A)^{-1} A^T b$$

Where A^T is the transpose of metrix A, so we can write, $\begin{bmatrix} Vx \\ Vy \end{bmatrix}$=

$$\begin{bmatrix} \sum_i I_x(p_n)^{\,2} & \sum_i I_x(p_n) I_y(p_n) \\ \sum_i I_x(p_n) I_y(p_n) & \sum_i I_y(p_n)^{\,2} \end{bmatrix}^{-1} \begin{bmatrix} -\sum_{\mathrm{i}} I_x(p_n) I_t(p_n) \\ -\sum_{\mathrm{i}} I_y(p_n) I_t(p_n) \end{bmatrix}$$

Here *n*=1 to *N*.

The matrix $A^T A$ is often called the structure tensor of the image at the point *p*.

Weighted window –
The plain least squares solution above gives the same importance to all N pixels p_n in the window. In practice it is usually better to give more weight to the pixels that are closer to the central pixel $\boldsymbol{p}$. For that, one uses the weighted version of the least squares equation [9],

$A^T WAV = A^T Wb$, We can write this equation in another form so we get:

$$V = (A^T WA)^{-1} A^T Wb$$

Where W is an *n*×*n* diagonal matrix containing the weights Wii = wi to be assigned to the equation of pixel p_n . That is, it computes:$\begin{bmatrix} Vx \\ Vy \end{bmatrix}$ =

$$\begin{bmatrix} \sum_{\mathrm{i}} w_i I_x(p_n)^{\,2} & \sum_{\mathrm{i}} w_i I_x(p_n) I_y(p_n) \\ \sum_{\mathrm{i}} w_i I_x(p_n) I_y(p_n) & \sum_{\mathrm{i}} w_i I_y\ (p_n)^{\,2} \end{bmatrix}^{-1} \begin{bmatrix} -\sum_{\mathrm{i}} w_i I_x(p_n) I_t(p_n) \\ -\sum_{\mathrm{i}} w_i I_y(p_n) I_t(p_n) \end{bmatrix}$$

The weight *wi* is usually set to a Gaussian function of the distance between p_n and *p*.

Lucas and Kanade and others implemented a weighted least squares (LS) fit of local first-order constraints to a constant model for v in each small spatial neighbourhood by minimizing [5].

$$\sum w^2(x,\ y)\ \ [\nabla I(x,\ y,\ t)\, v + \ It(x,\ y,\ t)]^{\mathbf{2}}$$

Where $w(x,\ y)$ denotes a window function that gives more influence to constraints at the center of the neighbourhood.

Compute Iterative LK at highest level
For Each Level *I* [7];

- Take flow *u*(*i*-1), *v*(*i*-1) from level *i*-1
- Up sample the flow to create u*(i), v*(i) matrices of twice resolution for level *i*.
- Multiply u*(i), v*(i) by 2
- Compute It from a block displaced by u*(i), v*(i)
- Apply LK to get u'(i), v'(i) (the correction in flow)
- Add corrections *u*'(*i*), *v*'(*i*) to obtain the flow *u*(*i*), *v*(*i*) at *I* th level, i.e., *u*(*i*)=*u**(*i*)+*u*'(*i*), *v*(*i*)=*v**(*i*)+*v*'(*i*)

Errors in Lucas-Kanade according to CeLiu [6] - causes of errors in this procedure are following:

- $\mathrm{A^TA}$ is easily invertible.
- There is not much noise in the image when particular assumptions are violated.
- Brightness constancy is not satisfied.
- The motion is not always small.
- There is some problem to move pixel position like its neighbors:
  –window size is too large
  –what is the ideal window size?

5. LUCAS KANADE FEATURE TRACKING

Lucas Kanade feature tracker was used in order to determine the movement of each sub target. The feature finding algorithm was also modified using eigen values in order to determine the trackable features according to Justin graham [11]. This involves keeping track of the highest eigen values over the search area. There are following steps for the feature tracker:

1. First compute the intensity for each pixel.
2. For each pixel position compute the gradient matrix and store an eigen value of matrix.
3. Store each pixel position in the score matrix S and separate the high scoring pixels by flag matrix F and region size k and flag region size f.
4. Take the top n eigen values and use those for the trackable features.

This algorithm effectively finds the edges of the image which contain the most information**.** The Lucas Kanade implementation could keep track of targets throughout affine transformations and occlusions. Although there are still some improvements to be made to handle occlusion and maintain stability [11].
Lucas Kanade tracker when the tracker is no longer able to maintain a good track on the target. The Lucas Kanade tracker does not require a particle filter.
Particle Filter - this particle system implementation, the weighting system will have two modes that can be activated at a time that is convenient for the tracking system. The first mode will take the final output of the Lucas Kanade tracker and use those as inputs to the particle system measurements [11].
The particle system is still going through all of the previously described steps which consist of:

- The next frame predicts the constant velocity motion model.
- Weight each particle based on its distance from the Lucas Kanade targeting output Using the current frame.
- The Gaussian random distribution for speed base.
- Resample the particles based on the weight.
- Repeat step first until complete.

The Lucas Kanade algorithm returns a high pixel difference score or a state of rapid change for the target.

6. CONCLUSION

In this paper, overview of some fundamental concepts related to motion estimation and optical flow has been provided. This paper presented the Lucas kanade method which is widely used in differential method for optical flow estimation and computer vision. The Lucas kanade method has a great potential for authentication technic. In future we would use the Lucas kanade method for feature extraction in intelligent video authentication.

7. REFERNCES

[1] Ja-Ling Wu, "Motion Estimation for Video Coding Standards", National Taiwan University.

[2] Chiara Ercole,"LPA-ICI approximation: applications to motion estimation and video denoising", University of Roma TRE, Year 2003-2004.

[3] HustonSJ, Krapp HG. Kurtz, Rafael. "Visuomotor Transformation in the Fly Gaze Stabilization System" Bielefeld University, Germany, Year 2008.

[4] Lihizelnikmanor, “motion estimation”, Cambridge University, year 2008.

[5] J.L.Barron and N.A.Thacker,”Computing 2D and 3D Optical Flow”, University of Manchester, year 2005.

[6] CeLiu, “Motion Estimation”, Microsoft Research New England year 2011.

[7] Mester R., Some step towards a unified motion estimation procedure. In proceedings of the 45th IEEE Midwest symposium on circuits and systems, 2002

[8] Viral Borisagar and Hiral Raveshiya, “Motion Estimation Using Optical Flow Concepts “, Gujarat University.

[9] Bruce D. Lucas “Image Matching by the Method of Differences”, Carnegie Mellon University, year 1984.

[10] Mariya Zhariy, “Introduction to Optical Flow”, Uttendorf 2005.

[11] Justin graham, “target tracking with Lucas kanade optical flow “, University of Texas, year 2010.