Optical Flow as a property of moving objects used for their registration

Wolfgang Schulz
Computer Vision Course Project
York University
Email:wschulz@cs.yorku.ca

1. Introduction

A soccer game is a real world situation where certain constants exist. There is one ball, 11 players in each team, two goals and distinct borders that convey certain meanings. With a knowledge of this system an approach to visually detect inherent objects can be made. For this project a small, well defined recognition task has been picked. Players and ball shall be registered on a basis of their distinct movement characteristic. Movement as a kind of motion in the real world relates to brightness variations when projected to a two-dimensional medium, like an image or a video sequence. The reverse process, namely estimating the motion field from a sequence of images, is called the optical flow estimation.

1.1. Motivation

The foremost idea was to analyze a "real world" problem using techniques of a particular area of computer vision. The visual area of interest is a scene of a soccer game, the task is to obtain visual information, as positions of ball/players, line positions, directions of movement etc. That information can later be used as input data for an information system which could be blended on the original scene. It could as well be used to create a three-dimensional reconstruction of the investigated scene. Another application would be the control of a camera that tracks the ball, a player or any other recognized object. The variety of visual stimulus and motion in a soccer game is quite huge. Concerning the scene movement of the audience, of players and referees, the geometry of the field, maybe weather influence can be considered. Concerning the camera there may be zooming, panning, sudden perspective variations or a combination of those. Taking that into regard, a technique to analyze motion has been chosen.

1.2. Related Research

Optical flow as a method to recover a motion field can be used to recover three dimensional motion of the visual sensor and the three dimensional surface structure. Other application areas are motion detection, object segmentation, time-to-collision measurement (3). Of the variety of techniques to compute optical flow two major classifications can be made in a) differential, gradient based and b) feature based methods. Methods based on the spatial and temporal gradients of the image intensity have been applied by (1,2). Optical Flow in combination with image registration has been used by (5). A comparison of different methods has been done by (3).

1.3. Hypothesis

Ball and player should be distinguishable on a basis of their movement characteristics. The Ball is predominantly moving in one direction, whereas players perform submovements (with arms and legs) into different directions, though having one main direction. Two or more pictures are needed for a motion analysis. Within a sequence of $n > 2$ pictures the objects should be trackable by using their flow vectors as a window replacement. Evaluating further pictures of a sequence might enforce robustness by statistically substantiating the results.

2. Technical Approach

The method to estimate optical flow used in this project is gradient based (1), using the flow constancy approach, as shown by (6,7). As this method makes use of no higher than first derivatives, it is less sensitive to noise (6). The optical flow constraint equation

$$E_x u + E_y v + E_t = 0$$

is one equation and one unknown. This is not sufficient to define the optic flow solution. In other words there is only one independent measurement (brightness) available from the image sequence at a point, whereas the flow velocity has two components (1).
An additional constraint, namely that of flow constancy is introduced. This assumes that the changes in brightness patterns varies smoothly in the image (1). So over some finite window W the flow should be constant, at least for a short duration (3). To find a (u,v) that satisfies the constraint equation

$$E_x u + E_y v + E_t = 0$$

the squared violation of this constraint with respect to the variable of interest is seeked

$$min(u, v)(E_x u + E_y v + E_t)^2$$

Now the term is differentiated with respect to the variables of interest, namely u and v

$$min(u, v) \sum_{W_i} \sum_{W_j} 2E_x(E_x u + E_y v + E_t) = 0$$

$$min(u, v) \sum_{W_i} \sum_{W_j} 2E_y(E_x u + E_y v + E_t) = 0$$

which yields two equations in two unknowns. Converting into matrix form gives

$$\begin{pmatrix} \mathbf{u} \\ \mathbf{v} \end{pmatrix} = \begin{pmatrix} \sum\sum {E_x}^2 & \sum\sum E_x E_y \\ \sum\sum E_x E_y & \sum\sum {E_x}^2 \end{pmatrix} \begin{pmatrix} -\sum\sum E_t \\ -\sum\sum E_t \end{pmatrix}$$

A typical size of the finite window W is taken (5x5), as proposed by (7).

To attenuate noise, that is amplified by the derivatives (E_x, E_y, E_t), spatial filtering as well as temporal filtering is applied to the images of the sequence. A Gaussian filter with unit standard deviation is used for filtering (6).

For the estimation of partial derivatives of the brightness (E_x, E_y and E_t) of an image point E(i,j,t) each point is convolved with a Sobel edge detector mask.

$$\mathbf{S}_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

$$\mathbf{S}_y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

$$g(x, y) = f(x, y) * h(x, y) = \sum_i \sum_j f(x-i, y-j) h(i, j)$$

The aperture problem is that over a finite window taken into consideration, a movement might not be recognized for it's speed or it's direction. To handle the aperture problem, the multi-resolution, multi scale principle of the Gaussian pyramids is applied. On the one hand fast movements can be recognized, on the other hand more input for the statistical evaluation is generated.

2.1. Problem Analysis

A sequence of pictures of a soccer game from the Television contains different kinds of motion that are not all directly related to the movement of the players or the ball. That is for example panning and zooming of the camera. To reduce the problem to one of detecting changes to a (more or less) fixed background, a sequence with a stationary camera has been hand shot.

The video sequence is transferred to single images (of size 320x240 with 255 greyscale levels). A number of at least two of these images is to be processed. For the first picture of the sequence, reference windows around the objects of interest (ball, players) are set manually. The optical flow will be calculated within these windows. The size, that will be set equally for all windows, should be appropriate to the size of the players. As the players are usually larger than the ball, the size will be appropriate for the ball.

Lowpass Gaussian Pyramids with a height of two are generated to have more evaluable data. Flow analysis of the pyramid represents fine to coarse movement. On the first level (original image size) fine movement is captured whereas higher levels resemble more coarse movement. A level of two has been chosen for the example sequences since the object window size on the first level is with 30 already quite small.

The optical flow is calculated within these windows with the flow constancy method. The output is written to pgm files, separately for U and V as well as *.dat files, that are readable by Matlab. Both give a visual representation of the flow values for each pixel.

To evaluate the flow results in terms of object registration, statistical distributions are calculated. Usual methods are applied as: min, max, the four moments (mean, variance, m3, m4), skewness, kurtosis. These values are written to *.dat files for visual analysis in Matlab. Finally the appropriate statistical functions have to be chosen or expanded to yield a proper differentiation between the objects.

So far two subsequent pictures are necessary. Using more than two pictures of a sequence requires the recentering of the object windows with regard to the objects, since the objects have presumably moved from one frame to the next. As the optical flow vector $\binom{u}{v}$ has already been calculated, it can be used for the replacement. To track the displacement of the objects in the subsequent pictures a finite displacement method as proposed in (6) is supposed to be used.

To have a measure of quantifying the results of the algorithm, each window is flagged with a property "player" or "ball". The results of the statistical evaluation are then compared to those demands.

2.2. Implementation

The Implementation is done in C. Input Images are read and written in pgm binary format. The images can have any size but have to have a fixed size related to each other. Parameters are the filenames filename001, filename002 ... Further parameters set positions of the objects in the first frame, size of all objects, method of statistical output etc.

example: optflow img1.pgm img2.pgm ...

The main steps of the algorithm are
(1) read images and create pyramids
(2) filter each image of the sequence with a Gaussian filter
(3) filter each image of the sequence along the temporal dimension with a Gaussian Filter
(4) calculate the optical flow values u and v for each pixel in the object windows
(5) evaluate the kind of object depending on intrinsic flow parameters (by making use of statistical evaluation methods) and write results to file
(6) reposition the window according to the calculated optical flow
(7) repeat with the next object, pyramid-level, sequence-image-pair (go to 4)

3. Empirical evaluation

Two sequences each with ten pictures are used as input. Objects are hand-set with respect to center and type. That yields twenty tries to register the objects. The first frame has the ideal window positions, since these are hand set. The subsequent pictures will reposition the windows or regions of interest to the new estimated centers. That will bring results depending on ideal or less ideal mappings of objects and regions of interest.

3.1. Experimental Design

As input for the algorithm video sequences have been shot on the York University Campus. Two scenes where ball and players have a spatial distance have been chosen to be an appropriate input. Sequences with occlusion have not been regarded.
Center coordinates of players and ball were hand-selected (2) and given as parameters to the program. The size of the region around the objects is given as

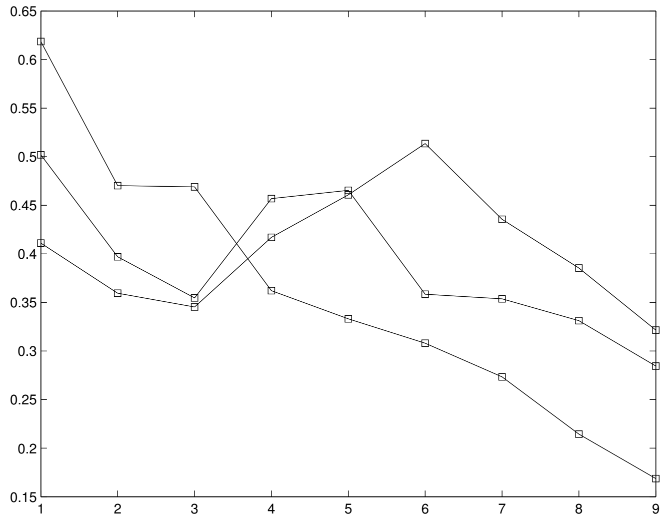

Figure 1. sequence1, 10 pictures: b) mean values, good results up to image 3, but then problems due to inproper tracking

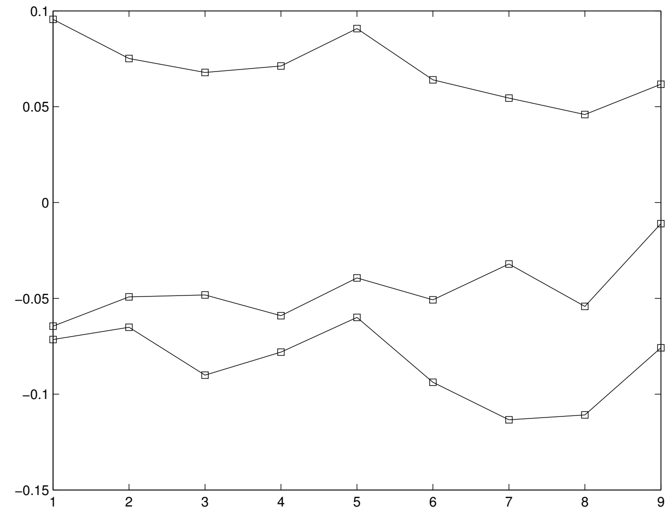

Figure 2. sequence2, 10 pictures: b) mean values, good results over the whole sequence

another parameter. The algorithm is going to register the object inside the window as a player or a ball. Afterwards the results can be compared to a manually set registration.

3.2. Results

The optical flow values as of u and v give information about speed and distribution of movement inside the marked regions of interest. The tracking does not yet work satisfying. This is due to the fact that it only relies on the calculated vector based on u and v of the first pyramid level. This vector yields a proper direction of the object's movement, but lacks proper distance. A method of finite displacement (6) would approximate the movement distance. A consideration of u,v from a higher pyramid level would give further support for proper displacement.

Comparing the statistical functions, the following have been found useful for the purpose of yielding differences between ball and player (see Fig. 1,2,3,4):
a) maximum positive value subtracted by the unsigned max. negative value: abs(max) - abs(min)
b) mean (or first moment) of values (Fig. 1,2)
c) mean of positive values subtracted by unsigned mean of negative values: pos.mean - abs(neg.mean) (Fig. 2)
It shows that the most robust differentiation is given by c). The diffPn() function filters the zero values, so that they do not have an influence on the height of the pos. and neg. mean distributions. Depictingly that is to put all positive values on a scale, to put all negatives values on the other scale and regard the tip of the scales. The player values balance the scales, whereas the ball values pin down the scale on either side.

Assuming that the objects are correctly tracked, the diffPn() function as a distinction of ball and players gives quite good results (see Fig. 3,4).

The figures show a mean calculated from the values of pyramid level 1 and 2. This mean is more significant than the values of each level.

In this project just the relative differences of "ball" and "player" flow results have been considered. A further step would be to have a threshold between those two regions. The results show that with the given methods it is difficult to name a fixed threshold.

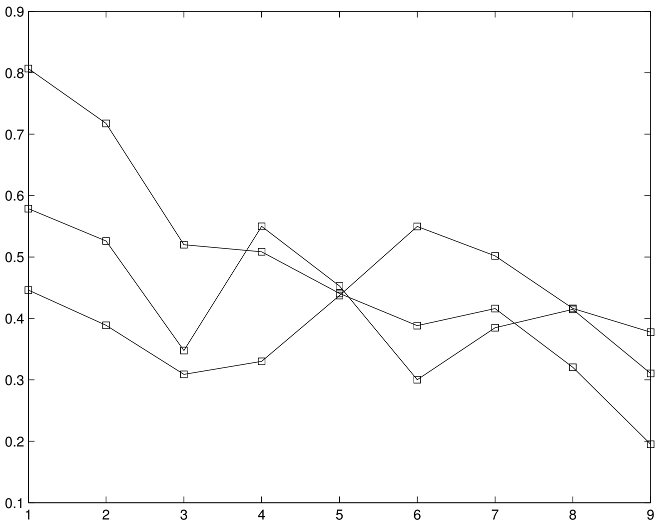

Figure 3. sequence1, 10 pictures: c) pos.mean - abs(neg.mean), good results up to image 3, but then problems due to inproper tracking

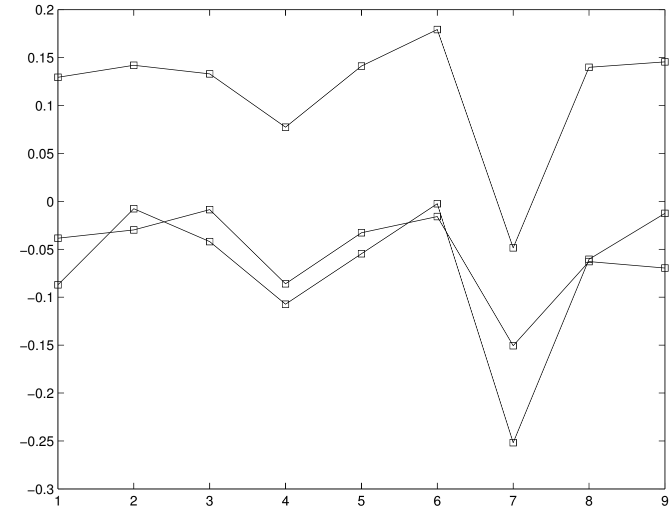

Figure 4. sequence2, 10 pictures: c) pos.mean - abs(neg.mean), good results over the whole sequence

Distribution of spatial support of non-zero flow values has not been considered as a measure.

4. Conclusion

The algorithms described above have been implemented and tested with exemplary extracts of the hand shot sequences. The results of the optical flow computation have shown to be a useful measure of differentiation of the objects of interest. The methods found for differentiating very specific objects as ”ball” and ”player” can be more put it in more general terms as methods to differentiate between objects moving in a uniform fashion (ball) versus objects moving in weighted directions (players).

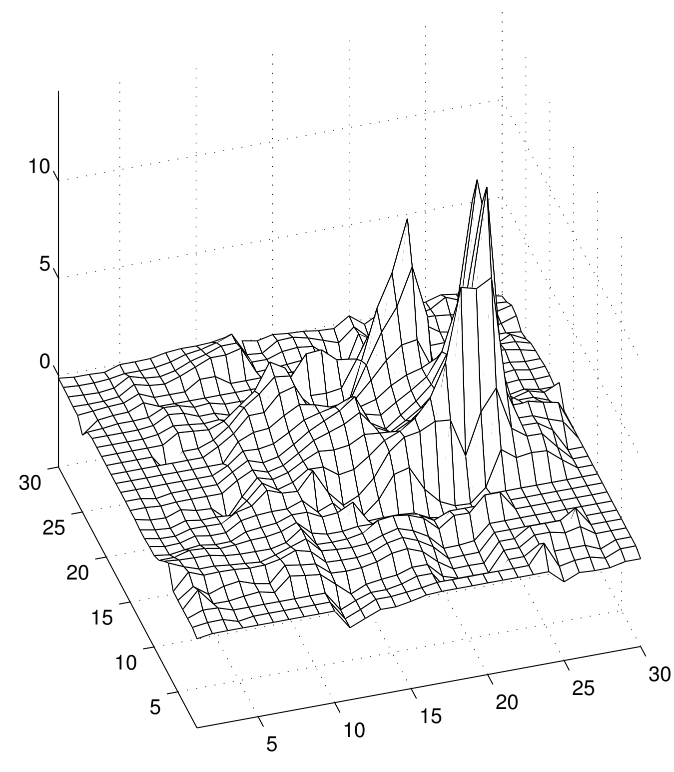

Figure 5. typical picture of the flow (here: u) of a ball

References

[1] B.K.P Horn, B.G.Schunck: “Determining optical flow”, *Artificial Intelligence*, 17, pp.185-203, 1981.

[2] B.D. Lukas and T.Kanade: “An iterative Image Registration Technique with an Application to

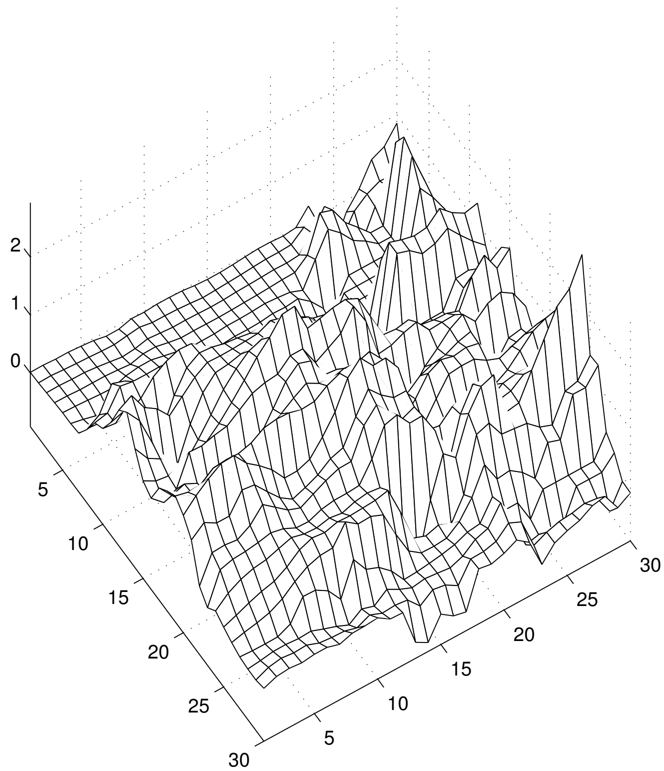

Figure 6. typical picture of the flow (here: u) of a player

Stereo Vision", *in Proceedings of 7th International Joint Conference on Artificial Intelligence*, pp.674-679, 1981

[3] S.S. Beauchemin, J.L. Barron: "Understanding Background Mixture Models for Foreground Segmentation" *Surveys(27), No. 3*, pp. 433-467, September 1995

[4] K.Y. Wong, M.E. Spetsakis: "Tracking, Segmentation and Optical Flow", *PDF*

[5] U.Neumann, S.You: "Integration of Region Tracking and Optical Flow for Image Motion Estimation", *something*

[6] R.Wildes "Computer Vision", *Lecture Material*, 2003

[7] E. Trucco, A. Verri: "Introductory Techniques for 3-D Computer Vision", *Prentice-Hall, Inc.*, ISBN: 0-13-261108, 1998

[8] E. Shah, R. Kumar: "Video Registration", *Kluwer Academic Publishers*, ISBN: 1-4020-7460-3, 2003