Reconstructing optical flow generated by camera rotation via autoassociative learning

Takashi TAKAHASHI
JSPS Research Fellow, University of Tsukuba,
305–8573 Japan
Takashi@Takahashi.com

Takio KURITA
Electrotechnical Laboratory,
305–8568 Japan
kurita@etl.go.jp

Abstract

We investigate methods to reconstruct the optical flow generated by camera rotation using autoassociative learning. A multi-layer perceptron is trained to reduce the dimensionality of flow data which are obtained from real image sequences while the camera is rotating against static scenes. After this learning, the perceptron is able to produce reconstructions of the flow removing the noises in the original flow data. It is also shown that robustness of reconstruction for noisy data is improved by two changes: introduction of confidence values of optical flow into the error function and application of an additional data correction method.

1 Introduction

The recovery of motion information from visual input is a crucial task for both of natural and artificial vision systems. For active vision systems allowed to move their cameras, estimation of the background flow produced by camera ego-motion is one of the important steps toward the detection of independently moving objects. In this paper, we investigate methods to recover the optical flow generated by camera rotation without information about rotation angles. Autoassociative learning of MLP(Multi-Layer Perceptron) is applied to the flow data in order to extract parameters relative to camera rotation. The resulting output of the perceptron reconstructs the flow removing noises. We also describe extensions of this method to obtain robust results against large noises. These methods are tested on the optical flow data obtained from real image sequences.

2 Optical flow generated by camera rotation

We consider the case that sequences of 2-D images are acquired by a camera mounted on the head of a human subject who stands still and is allowed to move his head up, down, right and left. To simulate such a system, let us define the coordinate system as shown in Fig. 1. The camera points toward the positive direction of Z-axis, and takes images projected on the image plane which is parallel to XY-plane and is located at a distance of the focal length f from the origin. Then the motion of the head-mounted camera can be approximated by a composition of two rotations around X- and Y- axes. Hence, on the ideal condition that there is no distortion of images such as lens distortion and there is no object moving independently, a component of the optical flow at position (x, y), $\boldsymbol{v}(x, y) \in R^2$, is given by

$$\boldsymbol{v}(x,y) = \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x \\ y \end{bmatrix} = -\begin{bmatrix} \frac{xy}{f} & f + \frac{x^2}{f} \\ -f - \frac{y^2}{f} & \frac{xy}{f} \end{bmatrix}\begin{bmatrix} \theta_X \\ \theta_Y \end{bmatrix}, \tag{1}$$

where θ_X and θ_Y denote angles of the camera rotation around X and Y axes, respectively. Figure 2 shows the examples of such optical flow.

If θ_X and θ_Y are known, it might be easy to estimate the optical flow evoked by camera rotation using the relation of Eq. (1) and to separate such components from other motion components. In most cases,

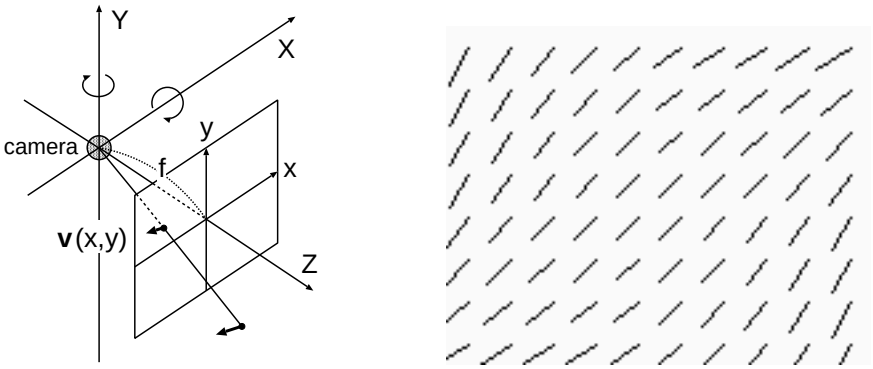

Figure 1: The coordinate system and an example of optical flow calculated using Eq. (1).

Figure 2: Examples of the optical flow generated by camera rotation: Left and Middle: the optical flow computed from a sequence of real images. Right: an example of the real image.

however, the rotation angles are unknown and must be estimated. Accordingly, we pursue a way to extract the parameters corresponding to the rotation angles only from observed flow data.

3 Autoassociative learning of camera rotation flow

3.1 autoassociative learning

In order to extract the parameters relative to the rotation of the camera, autoassociative learning is applied to a set of optical flow obtained from images while the camera is rotating. Autoassociative learning is one of the methods to reduce the dimensionality of multivariate data and extract a set of intrinsic parameters by using MLP. The output layer of MLP has the identical number of units to the input layer, and the bottleneck(middle) layer has smaller number of units than them. The MLP is trained to approximate the identity map of data, that is to say, to minimize the error between its output and the input. Hence, dimensionality-reduced data representation emerges as the output of the bottleneck-layer units, and the output of the MLP becomes a reconstruction of the input from this dimensionality-reduced representation. From a theoretical viewpoint, it is shown that a three-layer linear perceptron realizes the optimal dimensionality reduction which is equivalent to PCA(Principal Component Analysis)[1], and it is also known that a sandglass-type(typically, five-layer) MLP consisting of nonlinear units can perform nonlinear dimensionality reduction[2].

Let us consider the data vector whose components are optical flow: $\boldsymbol{\xi} = [\boldsymbol{v}^T(x_1, y_1), \ldots, \boldsymbol{v}^T(x_P, y_P)]^T \in R^{2P}$ where P denotes the number of flow components and (x_p, y_p) denotes the position where the p-th flow

components are obtained. If Eq. (1) holds for all components of this vector, $\boldsymbol{\xi} = A[\theta_X, \theta_Y]^T$ also holds for an appropriate $2P \times 2$ matrix A. This relation means that the flow data $\boldsymbol{\xi}$ can be generated by a linear transformation of two unknown parameters, θ_X and θ_Y, so that its intrinsic dimensionality is two. By using a three-layer linear perceptron which has two bottleneck-layer units, therefore, it is able to extract two principal parameters corresponding to these two parameters. As the consequence of dimensionality reduction, the resulting output of the MLP composes the reproduction of the optical flow removing noises.

3.2 Learning with ordinary MSE

We use a three-layer linear perceptron which consists of $N(= 2P)$ input units, two bottleneck-layer units, and N output units, respectively. Let $\{\boldsymbol{\xi}_t \in R^N\}_{t=1}^T$ denote the set of optical flow data, and let $\widetilde{\boldsymbol{\xi}}_t$ denote the output of the perceptron corresponding to $\boldsymbol{\xi}_t$. Then the error to be minimized by learning is the distance between $\boldsymbol{\xi}_t$ and $\widetilde{\boldsymbol{\xi}}_t$. In general, it is given by the MSE(Mean Squared Error),

$$E = \frac{1}{T}\sum_{t=1}^{T} ||\boldsymbol{\xi}_t - \widetilde{\boldsymbol{\xi}}_t||^2 = \frac{1}{T}\sum_{t=1}^{T}\sum_{p=1}^{P} ||\boldsymbol{v}_{tp} - \widetilde{\boldsymbol{v}}_{tp}||^2, \tag{2}$$

where $\boldsymbol{v}_{tp}$ and $\widetilde{\boldsymbol{v}}_{tp}$ denote the original and reconstructed flow component at the p-th position of the t-th data. In this paper, we call this autoassociative learning with respect to Eq. (2) "ordinary learning."

3.3 Learning with weighted MSE

In the case that the data include relatively small noises, ordinary learning might provide satisfying results. However, the flow data computed on real images usually include a large amount of noises, and the results are affected. In order to lessen the influence of noisy data on autoassociative learning, we investigate the weighted error as an alternative error:

$$E_W = \frac{1}{C}\sum_{t=1}^{T}\sum_{p=1}^{P} c_{tp}||\boldsymbol{v}_{tp} - \widetilde{\boldsymbol{v}}_{tp}||^2, \tag{3}$$

where c_{tp} represents the "confidence value" which weights the p-th flow component of the t-th data, and C is the normalization constant given by $C = \sum_{t,p} c_{tp}/P$. We call autoassociative learning with respect to Eq. (3) "weighted learning."

In order to measure the confidence value of each flow component, we compute the optical flow by the extended gradient method proposed by Simoncelli et al.[3]. By using this method, it is able to estimate the probability distributions of optical flow. In the experiments, we used the variance of the optical flow to measure the confidence value(see A).

3.4 Data correction

The optical flow computed on real images sometimes includes some outliers as shown in the middle of Fig. 2. Even if such large noises are included only in a portion of the flow, their influence might spread over the output of the MLP and could mislead the reconstruction. Since the reconstruction errors of such components tend to be exceptionally large, however, it is able to discriminate them from normal data and to correct the reconstructed flow. We adopt the following procedure to perform data correction:

1. For a given data $\boldsymbol{\xi} = [\boldsymbol{v}_1^T, \ldots, \boldsymbol{v}_P^T]^T$, calculate the error of each component $\boldsymbol{v}_p$, $e_p = ||\boldsymbol{v}_p - \widetilde{\boldsymbol{v}}_p||^2$.
2. Prepare a new input data $\widehat{\boldsymbol{\xi}} = [\widehat{\boldsymbol{v}}_1^T, \ldots, \widehat{\boldsymbol{v}}_P^T]^T$, provided that

$$\widehat{\boldsymbol{v}}_p = \begin{cases} \boldsymbol{v}_p & \text{if} \quad e_p < \theta \\ \widetilde{\boldsymbol{v}}_p & \text{otherwise,} \end{cases} \tag{4}$$

where $\widetilde{\boldsymbol{v}}_p$ is the output component corresponding to $\boldsymbol{v}_p$, and θ is the threshold of the error.

Table 1: Reconstruction errors for the training data and the test data.

	E		E_W	
	training	test	training	test
ordinary learning	91.2	104.2	50.0	55.4
weighted learning	99.0	106.6	43.3	47.8

Figure 3: Results of autoassociative learning: These figures depict the output of the perceptrons corresponding to the input data shown in the left of Fig. 2. Left: ordinary learning. Right: weighted learning.

3. Substitute $\widehat{\boldsymbol{\xi}}$ for $\boldsymbol{\xi}$ and calculate the output of the MLP again.

Even though the flow data includes outliers, the reconstructed output can be improved by iterating this procedure several times.

4 Experiment

This section shows the experimental results applying the proposed methods to the optical flow obtained from sequences of real images. Sequences of 2-D images were taken by a video camera mounted on a tripod, and the images were captured by a PC. The camera(Sony EVI–D30) can be rotated by remote control. Each sequence of images was captured during the camera was rotating at given constant angular velocities around X- and Y- axes. The size of each image was 80×60 pixels. From these sequences of the images, we computed the optical flow composed of $P = 10 \times 8$ components using the extended gradient method[3] based on Lucas and Kanade's approach[4]. In total, 224 flow data were obtained, and they were divided into two sets: a set of $T = 56$ data for learning and a set of 168 data for test. For the training data set, we trained two three-layer perceptrons by ordinary learning and weighted learning, respectively. Each perceptron consisted of $N = 2P$, two, and N linear units in the input, the bottleneck, and the output layer, respectively.

Table 1 indicates the errors calculated after learning, while Fig. 3 indicates the examples of the output which correspond to the training data shown in the left of Fig. 2. Although the improvement of E_W acquired by weighted learning seems to be small, there is a noticeable difference in the reconstructed flow in comparison with the result of ordinary learning. It is seen that the flow obtained by weighted learning is little affected by noises in the training data. Next, validity of the method of data correction was investigated. In the experiments, the threshold of Eq. (4) was chosen to be $\theta = \sum_{p=1}^{P} e_p/P$ for each data, and the data correction procedure was iterated five times. Figure 4 shows the examples of the output which correspond to the noisy test data shown in the middle of Fig. 2. Due to the background with low contrast(shown in the right of Fig. 2), the right-hand side of this flow data is full of all sorts of noises. In this case, the reconstruction without data correction fails to recover the correct flow, while the proposed method successfully recovers it removing the noisy components.

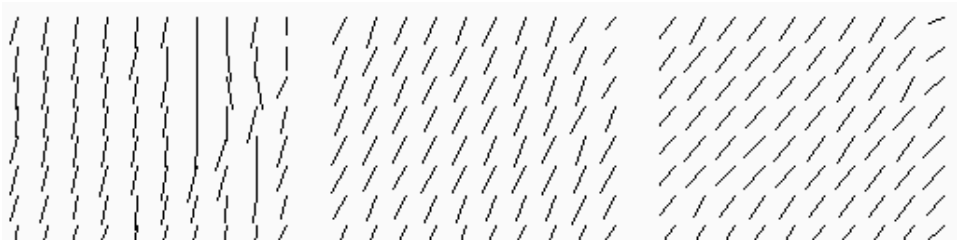

Figure 4: Effects of data correction. These figures depict the output of the perceptrons corresponding to the input data shown in the middle of Fig. 2. Left: ordinary learning. Middle: weighted learning. Right: weighted learning with data correction.

5 Discussion

We have investigated methods to reconstruct optical flow generated by camera rotation using autoassociative learning. Our approach has following advantages: (1) It provides estimation of the flow without calibrating the parameters of the camera. (2) It produces robust reconstructions for noisy data owing to the weighted learning method introducing the confidence value of optical flow and the data correction method. Future work will include extensions of this approach for detection and separation of moving objects from the background in situations of camera rotation.

Acknowledgments

The authors are grateful to Dr. Iwao SEKITA of Electrotechnical Laboratory for his insightful comments and kind cooperation. They also appreciate Mr. Yukifumi IKEDA of Saitama University for his invaluable aid in the experiments. This work was supported in part by Research Fellowships of the Japan Society for the Promotion of Science for Young Scientists.

References

[1] P. Baldi and K. Hornik, "Neural networks and principal component analysis: Learning from examples without local minima," *Neural Networks*, vol.2, pp.53–58, 1989.

[2] S. Usui, S. Nakauchi, and M. Nakano, "Internal color representation acquired by a five-layer neural network," *Proc. Int. Conf. Artificial Neural Networks*, vol.1, pp.867–872, 1991.

[3] E. P. Simoncelli, E. H. Adelson, and D. J. Heeger, "Probability Distributions of Optical Flow," In Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp.310–315, 1991.

[4] B. Lucas, and T. Kanade, "An iterative image registration technique with an application to stereo vision," In Proc. DARPA Image Understanding Workshop, pp.121–130, 1981.

A Calculation of the confidence value

Simoncelli et al.[3] investigated extensions of conventional gradient method which compute probability distributions of optical flow. By using this method, one can compute the mean vector $\boldsymbol{\mu}$ and the covariance matrix Λ for each flow component $\boldsymbol{v}$. Then $\boldsymbol{\mu}$ is the maximum likelihood estimate of $\boldsymbol{v}$, and Λ provides a two-dimensional uncertainty of the estimate. For computational simplicity, we have chosen to use a scalar c calculated by

$$c = \min(\lambda_1, \lambda_2) \tag{5}$$

where λ_1 and λ_2 are the eigenvalues of Λ^{-1}, as the confidence value.