MULTI-OBJECT TRACKING VIA HIGH ACCURACY OPTICAL FLOW AND FINITE SET STATISTICS

Marek Schikora, Wolfgang Koch

Fraunhofer FKIE
Dep. Sensor Data and Information Fusion
Wachtberg, Germany

Daniel Cremers

Technical University of Munich
Computer Science Department
Garching, Germany

ABSTRACT

In this work we present a novel method for tracking an unknown number of objects with a single camera system in real-time. The proposed algorithm is based on high-accuracy optical flow and finite set statistics. In this framework the target state is treated as a random vector and the number of possible objects as a random number, which has to be estimated correctly. We are able to deal with false alarms, clutter and object spawning. Since possible objects can appear or disappear in the scene we propose a probability model for these events, in order to obtain stable results in the case of missing detections. Additionally, we show how track labeling, based on color and state information, can improve the results. Since the method partly relies on color information, it can handle partial occlusion and is invariant to rotation and scaling. We verify the theoretical results on various scenes.

Index Terms— multi-object tracking, PHD-filter, optical flow, sequential Monte Carlo

1. INTRODUCTION

Multi-object tracking using a monocular system is a challenging but very important problem in many computer vision applications. The aim is to estimate the number and the state information of every object for each time step in an image sequence. The problem becomes challenging when the number of objects is unknown and variable. The *finite set statistics* (FISST) proposed by Mahler [1] are a systematic treatment for multi-object tracking with an unknown and variable number of objects. To reduce the complexity Mahler proposed an approximation of the original Bayes multi-target filter, the *Probability Hypothesis Density* filter (PHD). In [2, 3] it was shown that the PHD filter outperforms the classical approaches like Kalman Filter, standard particle filters and the Multiple Hypothesis Tracking. Algorithms based on the Joint Probabilistic Data Association filter (JPDAF) [4] tend to merge tracking results produced by closely spaced objects. This drawback cannot be observed, when using the PHD filter. In [5, 6] the authors use a PHD filter for multi object tracking, with the drawback, that only position information, gained from a detector, is used, so that the filter must estimate the velocity indirectly, which reduces the robustness of the filter. In this work we extend the classical PHD filter to deal with image data and velocity information gained from optical flow. For every object tracking task some kind of measurement is needed. Using optical flow we directly obtain velocity measurements for every pixel. Unfortunately, this does not provide any information about the position of objects in the scene. Fortunately, recently there has been a rapid progress in the field of object detection strategies [7, 8]. Since not all of these strategies are able to run in real-time we will present a fast strategy for moving object detection based on optical flow. The main idea of our tracking algorithm is to run an object detector and additionally compute the optical flow information. This gives us two kinds of measurements, which can be used for a stable and robust multi-object tracking, sample results can be seen in Figure 1. This paper is structured as follows: Section 2 describes the main ideas of optical flow and the moving object detector. Section 3 introduces the PHD. The following Section 4 describes our proposed algorithm and Section 5 presents experimental results.

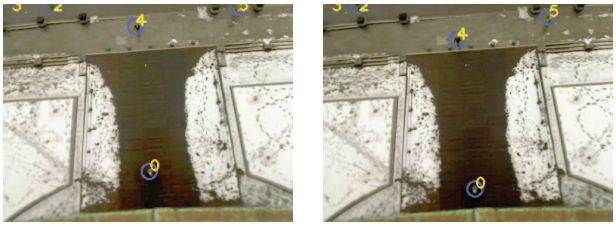


Fig. 1. Tracking an unknown number of people using high-accuracy optical flow. Tracking results as blue circles and person ID in yellow.

2. OPTICAL FLOW

The estimation of the optical flow between two images is a well-studied problem in low-level vision. A diverse range of

optical flow estimation techniques have been developed and we refer to the survey [9] for a detailed review. Taking into account the so-called Middlebury dataset [10] the discontinuity-preserving variational models based on Total Variation (TV) regularization and L^1 data terms are among the most accurate flow estimation techniques. Because of this fact, we will use in this context the estimation technique proposed by Werlberger et. al. [11]. To make this paper self-contained we briefly reflect their work.

For two input images $I_0, I_1 : \Omega \subset \mathbb{R}^2 \rightarrow [0, 1]$ the optical flow model can be stated as

$$\min_{\mathbf{u}} \left\{ \int_{\Omega} \sum_{d=1}^{2} |\nabla u_d| + \lambda |\rho(\mathbf{u}(\mathbf{x}))| \mathrm{d}\mathbf{x} \right\}, \tag{1}$$

with $\mathbf{u}(\mathbf{x}) = (u_1(\mathbf{x}), u_2(\mathbf{x}))^T$, $u_d : \Omega \rightarrow \mathbb{R}$, the free parameter λ to balance the relative weight of data and regularization term and $\rho(\mathbf{u}(\mathbf{x})) = \mathbf{u}(\mathbf{x})^T \nabla I_1(\mathbf{x}) + I_1(\mathbf{x}) - I_0(\mathbf{x})$ is the optical flow constrained equation. To improve the results and the accuracy the authors extend this approach using anisotropic Huber regularization. This approach has several benefits: firstly, the energy functional is convex, which leads to globally optimal solutions, and, secondly, the minimization can be scheduled in parallel, so that a real-time application can use the results without massive time drawbacks. Using this approach we can compute for every pixel $\mathbf{x} \in \Omega$ and each time step k the velocity $\mathbf{u}_k(\mathbf{x}) = (u_1, u_2)^T$ of this pixel.

2.1. Moving Object Detector

In this subsection we present a fast object detector, which is designed to detect moving objects. Given the flow field $\mathbf{u}_k$ at a given time step k, we can compute the probability, individually for every pixel, that it belongs to a moving object:

$$p_m(\mathbf{x}) = 1.0 - \exp\left(-\frac{1}{2} \frac{(\|\mathbf{u}((x))\|_2 - \mu)^2}{\sigma^2} \right). \tag{2}$$

Here μ and σ correspond to a normal distribution indicating that a pixel does not move. Using a stationary camera μ would be 0. Using a flying platform with downward-looking camera μ would correspond to the actual velocity of the platform. A typical value for σ in our experiments is 0.5. Given this probability image $p_m : \Omega \rightarrow [0, 1]$ we can compute the center of gravity for every region with a high probability of movement. Examples of this detector can be seen in Figure 2.

3. FINITE SET STATISTICS

In a single-object system, the state and measurement at time k are represented as two random vectors of possibly different dimensions. However, this is not the case in a multi-object system. Here the multi-object state and multi-object measurements are two collections of individual objects and measurements. The number of these may change over time. Furthermore, there exist no ordering for the elements of the multi-object state and measurement. Using the theory proposed in [1], the multi-object state and measurement are naturally represented as random finite sets X_k and Z_k. For those the first moment, or probability hypothesis density, is the analog of the expectation of a random vector. The integral value of the PHD over a given region in state space leads to the expected number of objects within this region. We define $D(\mathbf{x}_k|Z^k)$ as the PHD associated with the multi-object posterior $p(X_k|Z^k)$ at a time step k, with Z^k denoting the accumulated measurement from the time steps 1 to k. The PHD filter consists of two steps: prediction and update. The prediction can be realized through the following equation:

Fig. 2. Moving object detection for different frames in the image sequence. Top row: smoothed probability image for pixel movement; bottom row: position measurement displayed as green points.

$$D(\mathbf{x}_k|Z^{k-1}) = b(\mathbf{x}_k) + \int \left[p_s(\mathbf{x}_{k-1}) p(\mathbf{x}_k|\mathbf{x}_{k-1}) + b(\mathbf{x}_k|\mathbf{x}_{k-1}) \right] D(\mathbf{x}_{k-1}|Z^{k-1}) \mathrm{d}\mathbf{x}_{k-1}, \tag{3}$$

where $b(\mathbf{x}_k)$ denotes the intensity function of spontaneous birth of new objects, $b(\mathbf{x}_k|\mathbf{x}_{k-1})$ describes the intensity function of the random finite set of objects spawned from the previous state $\mathbf{x}_{k-1}$. $p_s(\mathbf{x}_{k-1})$ is the probability that the object still exists at the time step k given its previous state $\mathbf{x}_{k-1}$, and $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is the transition probability density of the individual objects. The update equation can be written as

$$D(\mathbf{x}_k|Z^k) \cong F(Z_k|\mathbf{x}_k) D(\mathbf{x}_k|Z^{k-1}), \tag{4}$$

$$F(Z_k|\mathbf{x}_k) = 1 - p_D(\mathbf{x}_k) + \sum_{\mathbf{z} \in Z_k} \frac{p_D(\mathbf{x}_k) p(\mathbf{z}|\mathbf{x}_k)}{\lambda c(\mathbf{z}) + \int p_D(\mathbf{x}_k) p(\mathbf{z}|\mathbf{x}_k) D(\mathbf{x}_k|Z^{k-1}) \mathrm{d}\mathbf{x}_k}, \tag{5}$$

with $p_D(\mathbf{x}_k)$ denoting the probability of the detection of the state $\mathbf{x}_k$. Furthermore, $p(\mathbf{z}|\mathbf{x}_k)$ is the measurement likelihood,

$c(\mathbf{z})$ the probability distribution for every clutter point and λ is the average number of clutter points per scan.

4. MULTI-OBJECT TRACKING

We implemented the theory described in the last section with a sequential Monte Carlo method, also known as particle filter. The first attempt to implement this technique for a tracking system, using the random finite set theory, was presented in [12]. In the following the state of an individual object will be represented by $\mathbf{x}_k \in \mathbb{R}^4$, with two random entries for the position and two random entries for the velocities. Each measurement $\mathbf{z}_k \in \mathbb{R}^4$ is represented analogous. For the sake of simplicity we assume that the object motion model of each target is linear with a constant velocity. Since we use a high-accuracy optical flow with a high frame rate (e.g. 30 fps) we do not need a more complicated motion model in our experiments. With this the object state prediction can be written as:

$$\mathbf{x}_k = \begin{pmatrix} \mathbf{I}_2 & \Delta T \mathbf{I}_2 \\ \mathbf{0}_2 & \mathbf{I}_2 \end{pmatrix} \mathbf{x}_{k-1} + \mathbf{s}_k, \tag{6}$$

with $\mathbf{s}_k$ a zero mean Gaussian white process noise, ΔT the time difference between step k and $k-1$. $\mathbf{I}_2$ denotes the identity matrix for two dimensions and $\mathbf{0}_2$ a 2x2 matrix with zeros. Using the particle filter we can model the birth process $b(\mathbf{x}_k)$ as a uniformly distributed set of new particles with small weights. The likelihood function is given by:

$$p(\mathbf{z}|\mathbf{x}) = \frac{1}{(2\pi)^2 |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{z}-\mathbf{x})^T \mathbf{\Sigma} (\mathbf{z}-\mathbf{x})\right), \tag{7}$$

with $\mathbf{\Sigma}$ the covariance matrix of the measurement noise.

At every time step k we have $\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{L_k}$ as a particle-based approximation of the PHD. The prediction, update and resampling for every time step and new measurements is done following the work in [12]. This gives us a particle cloud. To estimate the correct object states from this cloud we have to perform a clustering. In our experiments we use adaptive resonance theory (ART) clustering [13] , which estimates the number of clusters automatically, with a distance parameter as predefined user input. With this kind of clustering we are robust against estimation errors in the number of objects.

To establish an individual object trajectory we have to label each object correctly. We use two kinds of information: object state and the color distribution of the object. For both information we will use a likelihood-type function measuring the confidence that two objects from consecutive time steps $k-1$ and k are identical. The values of these likelihoods will be in the range of 0 (not identical) and 1 (identical). Let us assume that m is an object from the time step $k-1$ and n is a object from the time step k: then we can predict the object state of m using (6), so that we get $\tilde{m}$. The distance is defined as $d(\tilde{m}, n) = \|\mathbf{x}^{\tilde{m}} - \mathbf{x}^n\|_2$, with $\mathbf{x}^{\tilde{m}}$ and $\mathbf{x}^n$ the state vectors of the objects $\tilde{m}$ and n. The likelihood function is then

$$L_{\text{state}}(m,n) = \exp\left(-\frac{(d(\tilde{m},n))^2}{2\sigma_d^2}\right), \tag{8}$$

with σ_d the standard deviation of the distance information.

The likelihood function for the color measurement is based on the idea of similarity measures on color histograms, which has the benefit to be robust against non-rigidity, rotation and partial occlusions [14]. Suppose that the distribution is discretized into η bins. The color histogram $\mathbf{p}(\mathbf{x}) = \{p(\mathbf{x}^{(c)})\}_{c=1,\ldots,\eta}$ at position $\mathbf{x}$ is calculated as

$$p(\mathbf{x}^{(c)}) = f \sum_{\mathbf{x}_j \in \mathcal{N}(\mathbf{x})} g\left(\frac{\|\mathbf{x}-\mathbf{x}_j\|}{\alpha}\right) \delta(h(\mathbf{x}_j) - c). \tag{9}$$

In (9) f is a normalization factor, α is the scaling factor, $\mathcal{N}(\mathbf{x})$ denotes the neighborhood of pixel $\mathbf{x}$, δ is the Kronecker delta function and $g(.)$ is a weighting function given by

$$g(r) = \begin{cases} 1-r^2, & r < 1 \\ 0, & \text{otherwise} \end{cases}. \tag{10}$$

$h(\mathbf{x})$ is a function, which assigns the color at location $\mathbf{x}$ to the corresponding bin. To measure the similarity between two color distributions, which are denoted by $\mathbf{p}(\mathbf{x}) = \{p(\mathbf{x}^{(c)})\}_{c=1,\ldots,\eta}$ and $\mathbf{q}(\mathbf{x}) = \{q(\mathbf{x}^{(c)})\}_{c=1,\ldots,\eta}$, we use the Bhattacharyya coefficient. Let $\mathbf{p}_{\tilde{m}}$ and $\mathbf{q}_n$ be the color distribution of the objects $\tilde{m}$ and n, then the likelihood is:

$$L_{\text{color}}(m,n) = \sum_{c=1}^{\eta} \sqrt{p_{\tilde{m}}^{(c)} q_n^{(c)}}. \tag{11}$$

The likelihood that the objects m and n are identical, is a weighted sum over both likelihoods

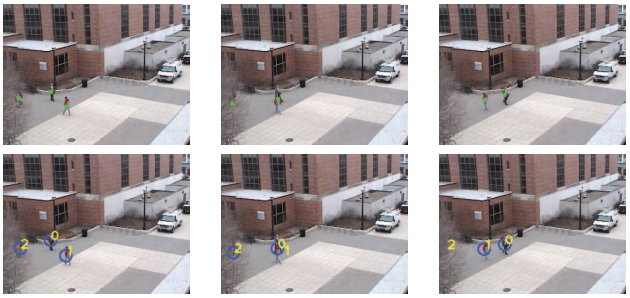
$$L(m,n) = w_p L_{\text{state}}(m,n) + w_c L_{\text{color}}(m,n). \tag{12}$$

Using this measurement (12) we can compute the similarity between every object from the time step $k-1$ and every object from the time step k. If the measurement exceeds a threshold value, then the objects are labeled as identical. If a new object does not match any other object from the previous time step, then a new object is added to the database. Objects that are not supported by new measurements over the time are deleted from the database, assuming that the object has left the scene.

5. RESULTS

In this section we present experimental results of our tracking algorithm. The image sequence used in Figure 3 was published in [15]. The top row of it shows the position measurement. In frame 61 the motion field of two persons merges (c.f. Figure 2), so that the detector measures only one moving object for a couple of frames. Because of the proposed labeling

and the PHD filter we are able to track and label both persons correctly when the motion fields splits again. This can be seen in the bottom row. The center of the blue circle corresponds to the position information gained by the clustering step after the particle update. The radius of this circle is fixed and only used for presentation. The yellow number is the ID of a person. In frame 86 the ID of person 2 is still displayed to indicate the last known position of this person. For this scene we had a hand-labeled ground truth. The mean position error between the proposed algorithm and the ground through lies by 2.68 pixel with a standard deviation of 1.5 pixel. Tracking



frame: 41 61 86

Fig. 3. Tracking result. Top row: position measurement displayed as green points in the image sequence. Bottom row: Position estimation of every object plotted as blue circle with fixed radius and id-number of every object in yellow.

and labeling results for a different scene can be seen in Figure 1. The image sequence used here was published in [16]. The challenge in this scene lies in the high false alarm rate of about 5-10%, which comes from additional noise produced through snowfall. Nevertheless, we could track and label all persons in this scene correctly. We computed the achieved runtime as frame rate means from the individual runtimes for each frame in the scenes from the Figures 3 and 1: 500 particles 61.74fps, 1000 particles 54.43fps and 1500 particles 31.85fps. The results were computed on a Intel Q8220 Qaud Core CPU with 4GB RAM using a single core implementation.

6. CONCLUSION

In this work we presented a novel multi-object tracking algorithm based on optical flow information and finite set statistics. Additionally we demonstrated that track labeling improves the results, in the case of occlusion and merged or missing detections (c.f Figure 3). The proposed multi-object tracking algorithm is able to deal with false alarms and clutter, so that a simpler detection strategy, based on velocity measurements, could be used. Furthermore we showed that the algorithm is able to track and label a unknown number of objects correctly. By combining an efficient implementation of the tracking algorithm with an optical flow running on the GPU, we were able to track and label objects in real-time. A lower processing time for the tracking could be achieved through a parallel implementation of the particle filter on the GPU.

7. REFERENCES

[1] R. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Trans. AES*, vol. 4, no. 39, pp. 1152–1178, 2003.

[2] K. Panta, B.-N. Vo, S Singh, and A. Doucet, "Probability hypothesis density filter versus miltiple hypothesis tracking," *Proc. SPIE*, vol. 5429, pp. 284–295, 2004.

[3] R. Juang and P. Burlina, "Comparative performance evaluation of GM-PHD filter in clutter," in *FUSION*, July 2009.

[4] Y. Bar-Shalom, T.E. Fortmann, and M. Scheffe, "Joint probabilistic data association for multiple targets in clutter," in *Conf. on Information Sciences and Systems*, 1980.

[5] Y. Wang, J. Wu, A. Kassim, and W. Huang, "Tracking a variable number of human groups in video using probability hypothesis density," in *ICPR*, 2006.

[6] E. Maggio, M. Taj, and A. Cavallaro, "Efficient multi-target visual tracking unsing random finite sets," *IEEE Trans. on TCSVT*, vol. 18, no. 8, pp. 1016–1027, 2008.

[7] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[8] M. Schikora, "Global optimal multiple object detection using the fusion of shape and color information," in *EMMCVPR*, 2009.

[9] J. Weickert, A. Bruhn, T. Brox, and N. Papenberg, "A survey on variational optic flow methods for small displacements," *Mathematical Models for Registration and Applications to Medical Images*, pp. 103–136, 2006.

[10] S. Baker, D. Schastein, J.P. Lewis, S. Roth, M.J. Black, and R.Szeliski, "A database and evaluation methodology for optical flow," in *ICCV*, 2007.

[11] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof, "Anisotropic Huber-L1 optical flow," in *BMVC*, London, UK, September 2009.

[12] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multi-target filtering with random finite sets," *IEEE Trans. AES*, vol. 41, no. 4, pp. 1224–1245, 2005.

[13] G.A. Carpenter and S. Grossberg, "ART 2: Self-organizing stable category recognition codes for analog input patterns," *Applied Optics*, vol. 26, no. 23, pp. 4919–4930, 1987.

[14] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter," *Image and Vision Computing*, vol. 21, no. 1, pp. 99–110, 2002.

[15] J. Davis and V. Sharma, "Background-subtraction using contour-based fusion of thermal and visible imagery," *Computer Vision and Understanding*, vol. 106, no. 2–3, pp. 162–182, 2007, IEEE OTCBVC WS Series Bench.

[16] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool, "You'll never walk alone: Modeling social behaviour for multi-target tracking," in *ICCV*, 2009.