Advanced Image Tracking Approach for Augmented Reality Applications

Ievgen M. Gorovyi and Dmytro S. Sharapov
It-Jim
1 Konstitucii Sqr., Kharkiv 61045, Ukraine
ceo@it-jim.com, sharapov@it-jim.com

Abstract — **augmented reality is popular and rapidly growing direction. It is successfully used in medicine, education, engineering and entertainment. In the paper, basic principles of typical augmented reality system are described. An efficient hybrid visual tracking algorithm is proposed. The approach is based on combining of the optical flow technique with direct tracking methods. It is demonstrated that developed technique allows to achieve stable and precise results. Comparative experimental results are included.**

Keywords— augmented reality, marker, visual tracking, local features, optical flow, direct tracking.

I. Introduction

Augmented reality (AR) is extremely popular and fast growing field [1]-[4]. Its basic idea is based on coexistence of real and synthetic (computer generated) objects. Practical usefulness is that AR allows to make some things more intuitive for the user [5]. As a result, AR is applied in such fields as medicine [6], education [5], entertainment and many more.

Nowadays humanity uses mobile phones and tablets daily. This enabled a strong interest for AR applications directly on-device [3]-[4]. As a result, new software and hardware for AR applications are constantly developed [3]-[4].

It is known, that efficiency of AR is strictly related with robustness and precision of applied computer vision algorithms [1]-[2]. From algorithmic point of view, AR required precise camera pose estimation (6-degrees of freedom). Availability of this information allows to augment an additional content such as 3d model, image and video, text data, audio, etc.

Camera pose estimation is accomplished by means computer vision algorithms applied for object recognition. A typical AR object is called marker. It can be considered as a predefined image with known properties. And it localization within the camera frame gives the full information about camera position and orientation.

There are two general types of AR systems: marker-based and markerless [1]-[2]. The former relies on a binary markers on the scene which can be easily tracked [7]. However, markerless systems are more popular since some natural image features can be used for detection and tracking [1]-[4]. Typically, planar objects are used for this purpose.

Since real scenes can be challenging due to occlusions, varying geometry and illumination changes, there is an active research in AR tracking algorithms. In addition, the goal is real-time performance, which could be difficult, especially for mobile devices.

An initial step before image tracking is detection. Output of this procedure is marker location within the camera frame (marker corners). For binary markers the contours analysis is typically applied [1]-[2], [7]. While for image markers a common option is to use keypoint descriptors [8]-[9]. A good method for real-time performance is oriented and rotated brief (ORB) [10]. As for image tracking, a common way is to use the optical flow (OF) algorithm [11]-[12]. The approach estimates the drift of interesting pixels (keypoints) in adjacent camera frames. In order to make the image tracking more robust, so-called direct tracking methods are often applied [13]-[15]. A key idea here is based on iterative estimation of the transformation between the template and test images using the whole image patch. An efficient second-order minimization (ESM) algorithm demonstrated good performance and faster convergence with respect to the Gauss-Newton scheme [13]. However, the drawback of ESM is requirement for a high amount of iterations in the case of fast camera motion.

In the paper, we describe a hybrid algorithm based on OF and ESM methods. It is demonstrated that such combining allows to make the tracking more robust and fast. As a result, camera pose is estimated precisely giving a realistic effect of augmented content.

In Section II, main principles of AR system are described. In particular, geometry peculiarities, marker types and principles of data augmentation. Section III describes the developed hybrid algorithm. Experimental results are discussed.

II. Augmented Reality Principles

A. Projective Geometry

A key of every AR application is estimation of the camera pose. Let's consider mathematical background of a problem. It is known that 3D and 2D worlds can be related using the projection equation [16]

$$s \times m_i = P \times M_i \,, \qquad (1)$$

978-1-5090-6754-1/17/$31.00 ©2017 IEEE

where s is a scale factor, m_i, M_i are world and projected point coordinates respectively, P is projection matrix

$$P = K \times [R \mid t] = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix}. \quad (2)$$

Here K is intrinsic camera matrix, $[R \mid t]$ is an extrinsic matrix describing the orientation and translation of the camera. The former does not depend on the scene and determines the camera parameters, namely, (c_x, c_y) is a principal point, f_x, f_y are focal lengths expressed in pixel units. The latter represents an Euclidean transformation from a world coordinate system to the camera coordinate system [16].

Fig. 1 illustrates the principle of point projection onto the image plane

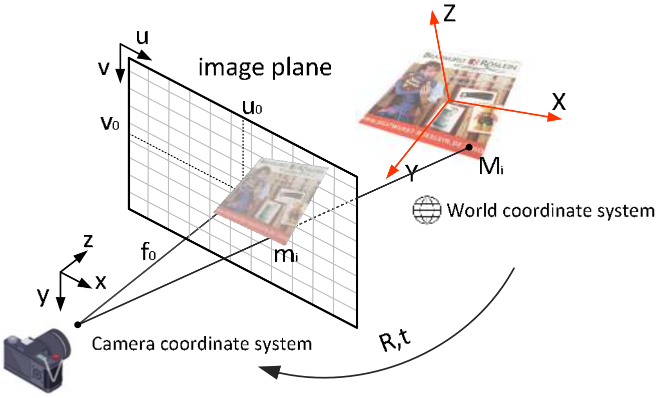


Fig. 1. Projective geometry.

One can see the planar object. An arbitrary point $M_i(X,Y,Z)$ is projected on the focal plane resulting in projection $m_i(u,v)$ .Thus, in the case of precise estimation of matrices K and $[R \mid t]$ Additional content can be augmented on camera frame in realistic way. One can show that it is enough to know exact 2D-3D correspondences for four pixels for a full camera pose reconstruction [1]-[2], [16].

B. Marker Detection and Data Augmentation

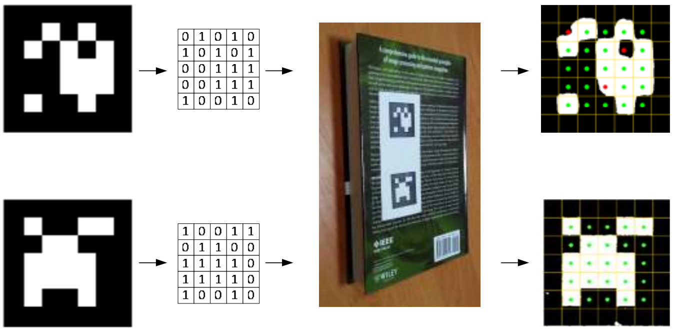
Let's consider how AR algorithm can be built. Firstly, one should emphasize that camera matrix K is estimation once during camera calibration procedure [1], [12], [16]. The situation is more challenging for extrinsic camera parameters estimation. It is known that relation between two arbitrary projections is described via the homography matrix [12], [16]

$$x' = \frac{x_{p'}}{z_{p'}} = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}, \quad y' = \frac{y_{p'}}{z_{p'}} = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}},$$

$$M(x', y', z') = H \times M_2(x, y, z), \; H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}. \quad (3)$$

Typically, $h_{33} = 1$ and, hence, homography is estimated up to a scale factor. As a result, 8 unknown matrix elements should be found from a set of point correspondences (at least 4 points). In order to perform consistent estimation, high amount of pixel pairs can be analyzed simultaneously using random sample consensus method (RANSAC) [1], [12].

Let's consider how different markers can be localized in camera frame. Fig. 2 illustrates how typical binary (fiducial) marker can be recognized.



0	1	0	1	0
1	0	1	0	1
0	0	1	1	1
0	0	1	1	1
1	0	0	1	0

1	0	0	1	1
0	1	1	0	0
1	1	1	1	0
1	1	1	1	0
1	0	0	1	0

Fig. 2. Fiducial marker recognition.

Firstly, contours are extracted from the input camera frame after thresholding [1]. Secondly, quadrilateral shaped contours are found. Finally, binary code matching is accomplished for marker candidates finalizing the marker recognition procedure. Apparently, that putting of binary markers into the scene is not practical. Therefore modern AR systems use much more convenient image markers. In this case marker detection in camera frame is performed using local feature descriptors. We used ORB descriptor in our AR system. Fig. 3 illustrates a principle of planar image marker detection.

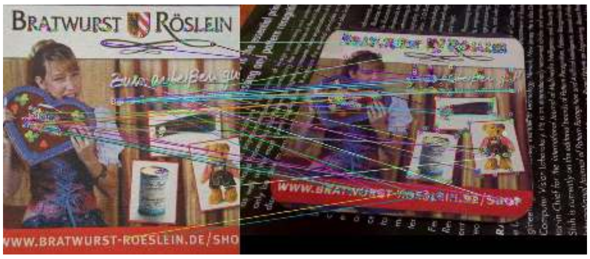


Fig. 3. Image marker detection example.

A predefined amount of keypoints is detected [1], [12]. Secondly, ORB descriptors are calculated for each keypoint. Finally, matching using Hamming distance metrics is accomplished [10]. One should emphasize that descriptors matching procedure is very important for the homography estimation [1]. Determined point-to-point correspondences are used for updating the camera pose.

Fig. 4 illustrates an example of data augmentation using estimated camera pose.

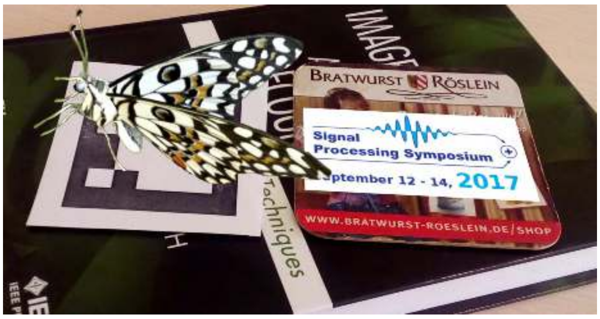


Fig. 4. Example of augmentation
(left – binary marker with 3d model, right – image marker with augmented image).

One can see an example of binary marker and augmented 3d model of butterfly (left). An example of image augmentation on the image marker is illustrated on the right side.

In general, marker detection algorithms can be applied consequently for camera frames after initialization. However, possibilities of local feature descriptors are limited due to varying scale and viewing angles [10], [12]. In addition, faster and more efficient tracking techniques can be used. Next subsection describes the developed tracking algorithm which is applied in frame-by-frame basis.

III. Experimental Analysis

This section contains a description of key steps of tracking algorithm and illustrates the experimental results.

A. Hybrid Tracking Algorithm Description

In order to keep augmentation stable and realistic, local image features should be properly tracked and kept within camera frames. We propose to combine the OF algorithm with ESM procedure.

The OF performs the estimation of the flow vector for a given interesting pixel using the local spatial window around the analyzed pixel

$$\begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_y I_x & \sum I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -\begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}, \tag{4}$$

where I_x, I_y are the horizontal and vertical image gradients, I_t is time gradient, i.e. difference between the frames, (u,v) are OF vector components. Summation in (4) is accomplished in a local window around the pixel of interest.

The challenge is that OF often fails when amount of keypoints is not enough or camera movement is too fast. In order to make image tracking more robust, we propose to refine the estimation using ESM algorithm. The key difference between local feature based and direct tracking is that for ESM the whole image patch transformation is estimated. Transformation parameters are evaluated based on minimization of a particular metric. For instance, sum of squared differences [13]

$$SSD = \sum_{x,y} (I_F^{warped}(x,y) - I_R(x,y))^2 \ . \tag{5}$$

Here I_R is a reference image (marker), I_F is the warped camera frame according to the homography of the previous frame

$$I_F^{warped} = I_F H_{F-1}^{-1} . \tag{6}$$

Thus, the perspective of previous camera frame can be used as initial estimation.

Let's consider the Taylor expansion

$$I_R = I_F^{warped} + J\Delta H + \frac{1}{2} Hes \cdot \Delta H \ , \tag{7}$$

where J is Jacobian, Hes is Hessian, ΔH is an unknown warping matrix (homography multiplier). The idea of ESM is based on approximation of the Hessian component [13] allowing to provide the faster algorithm convergence with respect to conventional Gauss-Newton scheme

$$\Delta H = -2(J + J_0)^+ (I_F^{warped} - I_R) \ , \tag{8a}$$

$$J_0 \approx J + Hes \ , \tag{8b}$$

where $(J + J_0)^+$ is a pseudoinverse extended Jacobian matrix, J_0 is Jacobian of identity warping [13]-[14]. As a result, such scheme allows to iteratively update the warping parameter ΔH .

We have integrated the proposed two-step tracking algorithm into mobile AR system. Fig. 5 illustrates a high-level scheme of the developed method. The initialization (marker detection) is accomplished using ORB descriptor. For marker tracking, OF algorithm is applied for consequent camera frames. Estimated pixel correspondences are used for calculation of the projection transformation. Secondly, warped camera frame is used as an input for iterative ESM algorithm providing additional refinement. Finally, camera pose is reconstructed [1]-[2]. The next subsection contains experimental analysis of the developed AR tracking algorithm.

B. Experimental Results

Let's analyze the performance of the method quantitatively. For this purpose, we have created a benchmark image sequence with known ground truth data (precise camera pose information).

The first important thing is the analysis of tracking capability and ESM convergence speed. Fig. 6 illustrates the amount of required iterations for each frame. One can observe that OF allows to substantially reduce the amount of ESM iterations. Also, for the ESM only the tracking algorithm fails in the middle of image sequence. In contrast, consequent application of OF and ESM allows to track the image marker and keep the moderate amount of ESM iterations for full convergence.

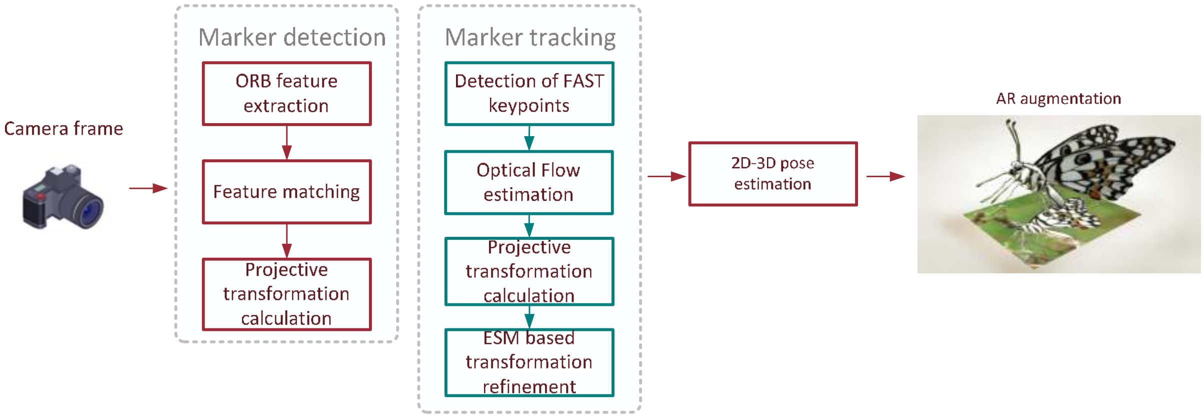


Fig. 5. Block-scheme of developed AR system.

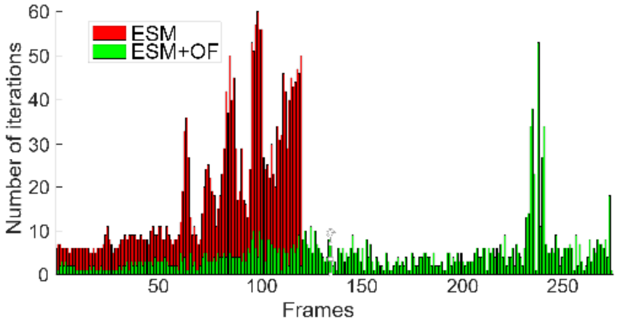


Fig. 6. ESM iterations.

Another practical example is camera pose reconstruction. Fig. 7 illustrates true camera pose and two reconstructed trajectories. Start and destination points are marked by arrows (Fig. 7). One can observe that combined marker tracking algorithm provides almost total coincidence with the ground truth. The maximum deflection in this case was not higher than several millimeters (green circles). For the case of OF usage for the image tracking, the accuracy is worse. The camera pose deflection was accumulated through the frames sequence resulting in around total 10 centimeters error (red circles). The test image marker was located at the origin (0, 0, 0).

Developed tracking algorithm was integrated into software development kit (SDK) intended for mobile devices (Android and iOS). Currently we are working on algorithm improvements and optimization.

IV. Conclusion

In the paper, a robust tracking algorithm for AR applications was proposed. Comparative analysis of OFLK and ESM approaches indicated on a potential of combining of these approaches. As a result, hybrid tracking method was developed. We have demonstrated that consequent application two proposed methods gives a possibility to achieve good results in terms of accuracy and speed. This is crucial for mobile devices and tablets with limized hardware capabilities. In the near future we are planning to fully adopt the developed algorithms for mobile devices.

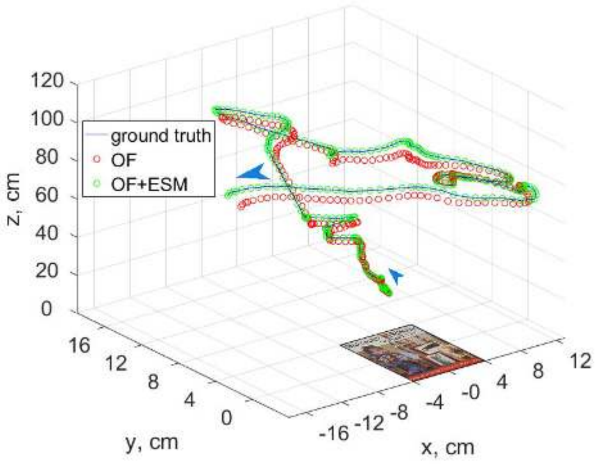


Fig. 7 Estimated camera pose.

References

[1] D. Baggio, S. Emami, D. Escriva, Mastering OpenCV with practical Computer Vision Projects. Packt Publishing, 2012.

[2] S. Siltanen,Theory and applications of marker based augmented reality. VTT Science 3, Espoo 2012.

[3] https://www.wikitude.com/products/wikitude-sdk/

[4] https://www.vuforia.com/

[5] P.Chen,X. Liu, W. Cheng, R. Huang. A review of using Augmented Reality in Education from 2011 to 2016. Innovations in Smart Learning.Part of the series Lecture Notes in Educational Technology, 2016, pp. 13-18.

[6] http://medicalfuturist.com/

[7] S. Garrido-Jurado, R. Muñoz-Salinas , F.J. Madrid-Cuevas , M.J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. Pattern Recognition. Volume 47, Issue 6, June 2014, Pages 2280–2292.

[8] H. Bay, A. Ess, T. Tuytelaars and Luc Van Gool. SURF: Speeded Up Robust Features. Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, 2008, pp. 346—359.

[9] D. Lowe. Distinctive image features from scale-invariant keypoints. International journal of computer vision, Vol. 60, No. 2, 2004, pp. 91-110.

[10] E. Rublee, V. Rabaud, K. Konolige and G. R. Bradski. ORB: An efficient alternative to SIFT or SURF, ICCV, 2011, pp. 2564-2571.

[11] J. Shi and C. Tomasi. Good features to track. In: IEEE Conf. on Computer Vision and Pattern Recognition, 1994, pp. 593-600.

[12] G. Bradsky and A. Kaehler, Learning OpenCV, O'Really Media Inc., 2008.

[13] S. Benhimane and E. Malis. Real-time image-based tracking of planes using efficient second-order minimization. IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 1, 2004, pp. 943-948.

[14] S. Benhimane and E. Malis. Homography-based 2d visual tracking and servoing. Int. J. Robot. Res. No. 26, 2007, pp. 661–676.

[15] S. Lieberknecht, S. Benhimane, P. Meier, N. Navab, Benchmarking template-based tracking algorithms. Virtual Reality 15(2-3):99-108. June 2011.

[16] R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University. Press, second edition, 2004.