

# Robust Estimation of Parameters for Lucas-Kanade Algorithm

Yih-Lon Lin<sup>1</sup>

**Abstract** The object tracking problem is an important research topic in computer vision. For real applications such as vehicle tracking and face tracking, there are many efficient and real-time algorithms. In this study, we will focus on the Lucas-Kanade (LK) algorithm for object tracking. In the standard LK method, sum of squared errors is used as the cost function, while least trimmed squares is adopted as the cost function in this study. The resulting estimator is robust against outliers caused by noises in the tracking process. Simulation is provided to show that the proposed algorithm outperforms the standard LK method in the sense that it is robust against the outliers in the object tracking problem.

**Keywords** Lucas-Kanade algorithm · Least trimmed squares

## 1 Introduction

The Lucas-Kanade (LK) algorithm was originally proposed by Lucas and Kanade in 1981 [1], which makes use of the spatial intensity gradient of the images to find a good match using a type of the Newton-Raphson iteration. The goal of the standard LK algorithm is to minimize the sum of squared errors (SSE) function between the template and the warped image region by adjusting the warping parameters.

Based on the theory of robust statistics, various types of cost functions were adopted in many practical applications such as regression. One of the main approaches to robust regression involves *M*-estimation [2-6]. The *M*-estimator is

---

<sup>1</sup> Yih-Lon Lin (✉)

Department of Information Engineering, I-Shou University, Taiwan  
e-mail: yllin@isu.edu.tw

found to be robust to outliers in the response variable, but turned out not to be resistant to outliers in the explanatory variables, called leverage points. In fact, when there are outliers in the explanatory variables, the method has no advantage over least squares. In the 1980s, several alternatives to  $M$ -estimation were proposed as attempts to overcome the lack of such resistance. Least trimmed (sum of) squares (LTS) is a viable alternative [7-10].

There are several robust LK algorithms proposed in following literatures. The  $M$ -estimators were adopted to LK models for tracking problems in [11-15]. The basic idea is to replace the cost function SSE by Huber or Hampel function so that the effects of outliers may be degraded. In [11], the authors generalized the LK approach to histogram-based tracking algorithm. It establishes a closer link between template matching and histogram-based tracking methods. Schreiber [12] proposed a novel variant of the LK algorithm for tracking bilaterally symmetric planar objects from a moving platform. This algorithm was capable to cope with any warping transformation and can be generalized for the case of objects possessing higher symmetry. According to analysis of data distributions, Senst *et al.* [13] proposed a robust local optical flow approach based on a modified Hampel estimator with robust characteristics. Fan *et al.* [14] proposed a robust template tracking with weighted active drift correction. The minimization of active drift correction is achieved by the inverse compositional algorithm, which consists of the tracking term and the drift correction term. In [15], a robust LK template matching algorithm was based on evidence which is accumulated over many frames. This algorithm described the drift-correcting and used robust weights that are being updated from frame to frame. The previous papers emphasized robust inverse composition algorithm using a robust Huber or Hampel cost functions. However, there are many other methods for robust LK algorithm such as robust measurement of ocular torsion [16], visual tracking and learning [17], and graph-based transductive learning [18].

The LTS estimator is highly resistant to leverage points, and is robust to outliers in the response. When we expect there to be some number of observations in the data

that we wish to put no weight in the modeling, LTS estimator is usually a good choice. The percentage of the data that we wish to put no weight is termed the trimming percentage or trimming parameter, and this parameter is usually pre-specified for the data at hand. In this study, the main idea of this study is to utilize the resistant property of the LTS estimators in the LK algorithm (LTS-LK) for object tracking problems when there are outliers caused by, for instance, noises and occlusions. The updating rules for the warping parameters using LTS-LK approach will be derived. A simulation is provided in this paper, for book-object tracking. The simulation result shows that the proposed method can effectively track the object when salt-and-pepper noises corruption.

The rest of this paper is organized as follows. In Section 2, we outline the standard LK algorithm. Section 3 presents the detail of the proposed LTS-LK method. A experimental results is provided in Section 4 to verify the performances of the proposed method. Finally, a conclusion is made in Section 5.

## 2 Lucas-Kanade Algorithm

Template tracking of an object in a video sequence is performed by extracting a template in the first frame and then finding the region which matches the template as close as possible in the surplus frames. The goal of LK method is to align a template image  $T(\mathbf{x})$  to an input image  $I(\mathbf{x})$ , where  $\mathbf{x}=[x \ y]^T$  is a column vector representing the pixel coordinates. Let  $\mathbf{W}(\mathbf{x};\mathbf{p})$  denote the parameterized set of allowed warps, where  $\mathbf{p}=[p_1 \cdots p_6]^T$  is a vector of warping parameters.

Here, we consider the following set of affine warps

$$\mathbf{W}(\mathbf{x};\mathbf{p}) = \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \end{bmatrix} = \begin{bmatrix} 1+p_1 & p_3 \\ p_2 & 1+p_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} p_5 \\ p_6 \end{bmatrix}. \quad (2.1)$$

The best match to the template in the new frame is found by minimizing the following SSE function

$$E = \sum_{\mathbf{x}} [I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x})]^2, \quad (2.2)$$

where  $\mathbf{W}(\mathbf{x}; \mathbf{p})$  is defined in (2.1) and the sum is performed over all of the pixels  $\mathbf{x}$  in the template image  $T(\mathbf{x})$ . Suppose a current estimate of  $\mathbf{p}$  is known and we wish to compute an appropriate increment  $\Delta \mathbf{p}$ . The minimization problem in (2.2) can then be converted to

$$\arg \min_{\Delta \mathbf{p}} \sum_{\mathbf{x}} [I(\mathbf{W}(\mathbf{x}; \mathbf{p} + \Delta \mathbf{p})) - T(\mathbf{x})]^2. \quad (2.3)$$

Performing a first-order Taylor series expansion on  $I(\mathbf{W}(\mathbf{x}; \mathbf{p} + \Delta \mathbf{p}))$  at  $\Delta \mathbf{p} = 0$  in (2.3) yields

$$\sum_{\mathbf{x}} \left[ I \left( \mathbf{W}(\mathbf{x}; \mathbf{p}) + \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} \right) - T(\mathbf{x}) \right]^2, \quad (2.4)$$

where  $\nabla I = [\partial I / \partial x \quad \partial I / \partial y]^T$  is the gradient of the image  $I$  evaluated at  $\mathbf{W}(\mathbf{x}; \mathbf{p})$ . The term  $\partial \mathbf{W} / \partial \mathbf{p}$  is the Jacobian of the warp. An approximate solution can be obtained by taking the partial derivatives of (2.4) with respect to  $\Delta \mathbf{p}$  and setting them to zero. It is easy to derive that this approximate solution is given by

$$\Delta \mathbf{p} = H^{-1} \sum_{\mathbf{x}} \left[ \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T [I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x})],$$

where  $H$  is Gauss-Newton approximation to the Hessian matrix

$$H = \sum_{\mathbf{x}} \left[ \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T \left[ \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right].$$

Finally, the parameters are updated as  $\mathbf{p} \leftarrow \mathbf{p} + \Delta \mathbf{p}$ , and the process is iterated until the estimate of parameters  $\mathbf{p}$  converge or the pre-specified stopping criterion is met.

### 3 Robust Lucas-Kanade Algorithm

Let  $X \subseteq \mathbb{R}^n$  and  $Y \subseteq \mathbb{R}^p$ . Suppose we are given the training set

$$S := \left\{ (x_q, d_q) \right\}_{q=1}^l \subseteq X \times Y.$$

In the following, we will use the subscript  $q$  to denote the  $q$ th observation. For instance,  $x_{qi}$  denotes the  $i$ th component of the  $q$ th input  $x_q \in \mathbb{R}^n$ ,  $q \in \underline{l}$ ,  $i \in \underline{n}$ . The residual (or error)  $e_{qk}$  at the  $k$ th component of the difference between the

desired output  $d_{qk}$  and the predicted output  $y_{qk}$  due to the  $q$ th observation is defined by

$$e_{qk} := d_{qk} - y_{qk}, \quad q \in \underline{l}, \quad k \in \underline{p}.$$

The LTS approach is to choose parameters that minimize the total sum of trimmed squared errors given by

$$\Psi_{total} = \frac{1}{2} \sum_{k=1}^p \sum_{q=1}^{l^*} e_{(q)k}^2 = \frac{1}{2} \sum_{k=1}^p \sum_{q=1}^l a(R(e_{qk}^2)) e_{qk}^2,$$

where  $l^* \leq l$  and the penalizing weight  $a(j)$  is defined by

$$a(j) := \begin{cases} 1, & 1 \leq j \leq l^*, \\ 0, & l^* < j \leq l. \end{cases}$$

Here,  $R(e_{qk}^2)$  denotes the rank of the residual  $e_{qk}^2$ ,  $q \in \underline{l}$  and  $e_{(1)k}^2 \leq \dots \leq e_{(l)k}^2$  are the ordered values of  $e_{1k}^2, \dots, e_{lk}^2$ . A popular choice of  $l^*$  is  $l^* = \lfloor l/2 \rfloor + \lfloor (n+1)/2 \rfloor$ , where  $\lfloor b \rfloor$  indicates the largest integer less than or equal to  $b$ . In the following, the term  $(l - l^*)/l$  is referred to as the trimming percentage.

To address the object tracking problems when there are outliers caused by, for instance, noises, we will use sum of trimmed squared errors instead of the  $l_2$ -norm. Consequently, the goal of the LTS estimator is to find the parameters  $\Delta \mathbf{p}$  of LK algorithm that minimize the cost function defined by

$$E = \sum_{\mathbf{x}} a(R([I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x})]^2)) \left[ I \left( \mathbf{W}(\mathbf{x}; \mathbf{p}) + \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} \right) - T(\mathbf{x}) \right]^2.$$

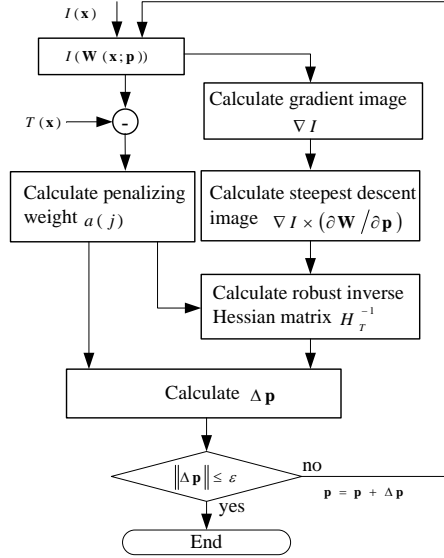
It is not difficult to derive that the variable  $\Delta \mathbf{p}$  can be calculated as

$$\Delta \mathbf{p} = H_T^{-1} \sum_{\mathbf{x}} a(R([I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x})]^2)) \left[ \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T [I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x})],$$

where the robust Hessian matrix is

$$H_T = \sum_{\mathbf{x}} a(R([I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - T(\mathbf{x})]^2)) \left[ \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T \left[ \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right].$$

The flowchart of the proposed LTS-LK algorithm is shown in Fig. 3.1.



**Fig. 3.1** Flowchart of LTS-LK algorithm.

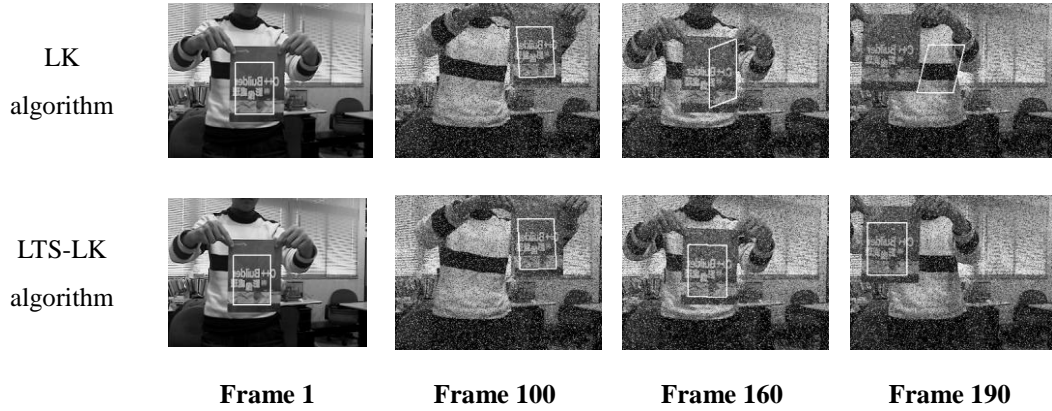
## 4 Experimental Simulation

In this section, we compare the performances of standard LK and the LTS-LK algorithms for object tracking. Emphasis is put particularly on the robustness against outliers. Simulation programs are implemented using Borland C++ Builder 6.0 running on Microsoft Windows 7, Intel Core 2 Quad CPU, and 4-GB RAM platform.

To demonstrate the robustness of the proposed algorithm, we present an object-tracking experiments, consisting of one “book-tracking” video sequences. In Example 1, the frames after the 100<sup>th</sup> of the “book” sequence are corrupted by salt-and-pepper noises in which a corrupted pixel has the intensity of 0 or 255. The trimming percentage used in our simulations is 5%. In the simulation, we set 20 as the maximal number of iterations and 0.0001 as the minimal change of  $\Delta \mathbf{p}$ . The

iteration stops when either of the two conditions is met.

**Example 1.** The "book" sequence is of size  $320 \times 240$  containing 250 frames in which the object to be tracked is the book. In this experiment, the initial bounding box size is  $60 \times 80$  pixels and contains about 6.25% of the frame size. Simulation results are shown in Fig. 4.1. The first and second rows show the results of the standard LK and the proposed LTS-LK algorithms, respectively. To demonstrate the robustness of LTS-LK algorithm, we randomly corrupt the video sequence with 20% noises, about 15,360 pixels, in the frames after the 100<sup>th</sup> frame. As observed, the standard LK algorithm fails to track the object in the 160<sup>th</sup> frame (the 3<sup>rd</sup> column), whereas our LTS-LK algorithm can effectively track the "book" object.



**Fig. 4.1** Results of "book-tracking" sequence. Salt-and-pepper noises are added after Frame 100.

## 5 Conclusion

The main issue of this paper is the robustness against outliers caused by noises in the video object tracking problems. The proposed robust algorithm adopts the LTS scheme for the LK algorithm. Experimental results show that the proposed LTS-LK algorithm outperforms the standard LK. It is noted that SSE minimization criterion

is used in standard approach to search the optimal parameters. However, it is known that the estimation of parameters based on least squares criterion is vulnerable to outliers.

**Acknowledgments** The research reported here was supported by the National Science Council, Taiwan, under grant no. NSC 101-2221-E-214-074 and NSC 102-2221-E-214 -049.

## References

1. Lucas, B. D. & Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. in *Proceedings of the International Joint Conference on Artificial Intelligence*, Canada, pp. 674-679.
2. Tukey, J. W. (1996). A survey of sampling from contaminated distributions. in *Contributions to Probability and Statistics*, Olkin, I., Ghurye, S., Hoefding, Madow, G. W. W. G., Mann, H. B., Eds. Stanford Univ. Press, Stanford, California, pp. 448-485.
3. Huber, P. J. (1964). Robust estimation of a location parameter. *Annals Math. Stat.*, 35, pp. 73-101.
4. Huber, P. J. & Ronchetti, E. M. (2009). *Robust Statistics*, 2<sup>nd</sup> ed. Hoboken, Wiley, New Jersey.
5. Hampel, F. (1968). *Contributions to the Theory of Robust Estimation*, PhD thesis, Univ. of California, Berkeley, California.
6. Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., & Stahel, W. A. (2005). *Robust Statistics: The Approach Based on Influence Functions*, John Wiley & Sons, New York.
7. Rousseeuw, P. J. (1985). Multivariate estimation with high breakdown point. in *Mathematical Statistics and Applications*, Grossmann W., Pflug G., Vincze, I., Wertz, W., Eds. Dordrecht, Holland, Reidel, pp. 283-297.
8. Tableman, M. (1994). The influence functions for the least trimmed squares and the least trimmed absolute deviations estimators. *Statistics & Probability Letters*, 19, pp. 329-337.
9. Giloni, A. & Padberg, M. (2002). Least trimmed squares regression, least median squares regression, and mathematical programming. *Mathematical and Computer Modelling*, 35(9), pp. 1043-1060.
10. Nguyen, T. D., & Welsch R. (2010). Outlier detection and least trimmed squares approximation using semi-definite programming. *Computational Statistics and Data Analysis*, 54(12) pp. 3212-3226.
11. Schreiber, D. (2008). Generalizing the Lucas-Kanade algorithm for histogram-based tracking. *Pattern Recognition Letters*, 29(7), pp. 852-861.
12. Schreiber, D. (2009). Incorporating symmetry into the Lucas-Kanade framework. *Pattern Recognition Letters*, 30 (7), pp. 690-698.
13. Senst T., Eiselein V., & Sikora T. (2012). Robust local optical flow for feature tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(9), pp. 1377-1387.
14. Fan, B., Du, Y., Zhu L., Sun J., & Tang, Y. (2011). A robust template tracking algorithm with weighted active drift correction. *Pattern Recognition Letters*, 32(9), pp. 1317-1327.
15. Fan, B., Du, Y., Zhu, L., Sun, J., & Tang Y. (2007). Robust template tracking with drift correction. *Pattern Recognition Letters*. 28(12), pp. 1483-1491.
16. Lee I. B., Choi B. H., & Park K. S. (2007). Robust measurement of ocular torsion using iterative Lucas-Kanade. *Computer Methods and Programs in Biomedicine*, 85(3), pp. 238-246.
17. Li, J., Wang, Y. & Wang, Y. (2012). Visual tracking and learning using speeded up robust features. *Pattern Recognition Letters*, 33 (16), pp. 2094-2101.
18. Zha, Y., Yang, Y., and Bi, D. (2010). Graph-based transductive learning for robust visual tracking. *Pattern Recognition*, 43(1), pp. 187-196, 2010.