Video stabilization algorithm for tunnel robots based on improved Kalman filter

He Liu¹, Shiyi Cui¹*

¹ School of Control and Computer Engineering, North China Electric Power University, Beijing, China

*Corresponding author’s e-mail: 1182227002@ncepu.edu.cn

Abstract. The application of tunnel robots in power tunnels is becoming more and more extensive. Due to various reasons, video jitter will inevitably occur during the inspection process, which will affect the real-time processing of subsequent images. In order to reduce the impact of video jitter and meet the requirements of real-time image processing, it is necessary to study fast video image stabilization methods. The traditional Kalman filtering algorithm has less computational cost, but it will cause a large estimation error when the system motion state changes. In this regard, the paper proposes an improved Kalman filtering algorithm, which changes the corresponding filter parameters by real-time estimation of a system in motion to reduce the estimation error. And this algorithm performs Harris corner extraction in the set feature dense area, and combines the PyrLK optical flow algorithm with the homography matrix to calculate the motion trajectory, which improves the calculation speed and accuracy of the algorithm. Experimental results show that this algorithm has better video stabilization effect and faster calculation speed compared with the traditional Kalman filtering algorithm, and can better meet the real-time image processing requirements in the inspection process.

1. Introduction

At present, tunnel robots are playing an increasingly important role in the inspection of power tunnel. The robot mainly uses the high-definition camera with visible light and infrared camera to take pictures of cables and tunnels, and recognizes the real-time status of cables and tunnels through image processing and multi-information fusion. During the inspection process, the videos taken by the robot will jitter, due to conditions such as hanging rail joints, rail foreign objects, abnormal emergency stop, etc. Video jitter will affect the subsequent image processing and real-time status recognition. Therefore, a fast and effective video stabilization method is needed to solve the video jitter problem that occurs during the inspection process.

Video stabilization methods can be roughly divided into three categories: mechanical image stabilization, optical image stabilization and digital image stabilization [1]. Compared with the previous two methods, the digital image stabilization method is more suitable for the environment of power tunnel as a software method. Digital video stabilization is generally divided into 2D video stabilization and 3D video stabilization. Since 3D video stabilization needs to be restored to the scene, it requires a large amount of calculation, takes a long time and has poor robustness. Currently, 2D video stabilization is mostly applied in practice. The 2D video stabilization process is generally divided into three parts: motion estimation, motion smoothing and motion compensation [2]. Morimoto and Chellappa [3] were the first to propose 2D video stabilization algorithm. They proposed to use similarity and homography matrix transformation to perform simple low-dimensional transformation processing of dither video,
which has good robustness, but cannot handle jitter video with large parallax. Liu et al. [4] proposed that video frames should be divided into multiple uniform grids. Under the unified optimization framework, path smoothing intensity could be adjusted adaptively through constraints such as discontinuity, cutting size and geometric distortion. It effectively solved the impact caused by large parallax, but the calculation required a large amount of work. Grandmann et al. [5] used optical flow method to track the extracted feature points for video stabilization. This method required a small amount of computation, but it was affected by the distribution of feature points.

Aiming at the problem of video jitter in the power tunnel inspection process, this paper proposes a video stabilization algorithm based on improved Kalman filter. In the motion estimation stage, a feature-intensive region containing more feature points can be manually selected for feature point extraction; and we combine the Pyramid optical flow method (Pyramid Lucas Kanade Optical Flow, PyrLK) and homography matrix transformation to calculate the motion trajectory according to the parameter information of extracted feature points; then we use the improved Kalman filter proposed in this paper to smooth the obtained motion trajectory in real time, and it can also be tracked in real time when the motion state changes suddenly; finally we obtain a stabilized video according to the smooth trajectory by using motion compensation on the jittery video.

2. Motion estimation

2.1. Feature Points Extraction

Harris corner detection algorithm is an algorithm proposed by Harris and Stephens, which has rotation invariance and high robustness to changes in brightness.

The principle of Harris corner detection is to use a small moving window to detect the change of gray value in various directions, and if there is a large change, it is judged as the corner position. For an image $I(x, y)$, the gray scale $E(u, v)$ generated by the displacement $[u, v]$ of moving the small window is changed to:

$$E(u, v) = \sum_{x,y} o(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

(1)

Among them, $I_x$ and $I_y$ represent the gradients of $I(x, y)$ in the x and y directions respectively, let

$$M = \sum_{x,y} o(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix}$$

set the corner response function as:

$$R = \det M - \alpha(\text{trace}M)^2$$

(2)

Among them, $\alpha$ is an adjustable parameter, which generally ranges from 0.04 to 0.15. The $R$ value is the condition for Harris algorithm to judge the characteristic point. If the $R$ value is greater than the set threshold range, the point is judged to be a feature point.

In order to adapt to the feature point extraction environment of the power tunnel, the algorithm in this paper adds freely selectable feature dense areas and Harris algorithm with distance constraints. Due to the detection process in the power tunnel, the area to be detected is concentrated in the middle of the video and contains more power equipment positions such as cables. The blank position contains a small number of detectable feature points and contains less information, but it occupies more calculation space. Therefore, setting the feature-intensive area can reduce the detection time of the whole area and improve the calculation efficiency. This algorithm has a small amount of calculation and can adapt to real-time requirements.

2.2. Motion trajectory extraction

In the process of extracting the motion trajectory, the method often used is to calculate the affine transformation matrix between two frames and iteratively calculate the motion trajectory. If there is an error in the calculation of the affine transformation matrix between two frames, the method will continue
to accumulate the error, Affecting the subsequent image stabilization process. In order to eliminate the influence of accumulated errors, this paper uses the PyrLK optical flow method to track the movement positions of the feature points, and calculates the homography matrix according to the movement positions of the feature points in the front and back frames to obtain the motion trajectory.

The key content of the PyrLK algorithm is: when searching for the corresponding feature point position in the detection neighborhood, first the image is reduced, then the offset distance of the corresponding feature point will also be reduced proportionally. And after finding the corresponding feature point in the reduced image, iterative calculation is performed at that position until the location information corresponding to the feature point in the original image is iteratively found.

The PyrLK algorithm will inevitably produce errors in the process of inter-layer transfer, so it will affect the feature point tracking between the final two frames. In order to reduce the impact of errors and extract the global motion vector, this paper uses a homography matrix to correct and reduce the influence of error, and uses Random Sample Consensus (RANSAC) that can eliminate mismatched points and calculate the optimal homography matrix.

The RANSAC algorithm first randomly selects a small number of feature points from the matching feature points in the preceding and following frames as initial sample points, and calculates its parameters to obtain an initial model. The feature points are classified by setting threshold conditions, and the feature points are divided into interior points and exterior points. The inner point refers to the correctly matched point that can be described by the data model, and the outer point refers to the mismatched feature point caused by noise, calculation or measurement errors, and does not conform to the data model.

After obtaining the optimal homography matrix $H_m$ of the $m$th frame and the $m+2$th frame, the motion trajectory $C_{m+1}$ can be extracted:

$$C_{m+1} = C_m H_m, m = 1, 2, ..., N \quad (3)$$

3. Motion smoothing and compensation

3.1. Kalman filter

In the power tunnel, it is necessary to provide timely feedback processing on the faults and problems found, which requires real-time video stabilization processing. Kalman filter is a dynamic data recursive processing algorithm. It only needs the state parameters of the previous moment to predict the current state parameters and reduce the influence of noise. And it has a small amount of data storage, which can perform real-time video stabilization well.

The essence of the traditional Kalman filter is based on the minimum mean square error as the best estimation criterion. The state space model of signal and noise is used to weight the motion parameters at the previous moment and the predicted values of the current motion parameters. The difference between the current measurement value and the current measurement value is calculated to obtain the weight $K$ at the next moment, and the recursive calculation is continuously performed to obtain the filtered parameters. The iterative update process of Kalman filter algorithm parameters is as follows:
3.2. Improved Kalman filter to smooth trajectory

In Kalman filtering, the dynamic noise covariance matrix $Q$ and the observation noise covariance matrix $R$ will directly affect the tracking and stability characteristics of the video stabilization filtering process. In the assumed ideal state, the initial system noise parameters can be measured and estimated. And it can remain unchanged in the motion state. Based on the assumption of traditional Kalman filtering, the state parameters can be estimated at any time, but in the actual movement process, the system noise will change with the change of movement state or the time, which is not a fixed value. So that the traditional Kalman filter cannot accurately estimate the parameter value under the state change. It may produce estimation errors, resulting in the smooth trajectory deviating greatly from the original trajectory.

In order to reduce the influence of the estimation error caused by the change of system noise and the possible over-clipping problem caused by large offset, the algorithm in this paper performs a real-time estimation of the current motion state after extracting the absolute path $CN$. If the current state contains less active scanning motion, that is a fixed scene state, $Q$ decreases and $R$ increases to ensure a better smoothing effect during the Kalman filtering process; on the contrary, if the current state contains more active scanning motion, that is, in a fast moving state, $Q$ increases and $R$ decreases, so that the filtering process can quickly track on the basis of ensuring stability and reduce the impact of estimation errors. It takes a sliding window with a size of $w=2m+1$ to perform real-time weighting calculation processing on the path parameters in the window, and calculates the deviation of its weighted displacement $Z_k$ from the weighted displacement $Z_{k-1}$ of the previous moment. If the deviation exceeds the set threshold for a certain number of times, it is judged that the current state contains more active scanning movement; if the deviation is always within the set threshold, or is always continuously exceeded, it is judged that the current state is a movement state that contains less active scanning. According to the real-time determined motion state, the value of $Q$ and $R$ would be adjusted in time.

4. Experiment Analysis

The video stabilization experiment is carried out on the video frame sequence containing the jitter component. This paper conducts comparative analysis from three aspects. First, we compare the average time consumption of feature point detection in the set feature dense area and global feature point detection on the video frame sequence, then we compare the effects of the improved Kalman filter and the classic Kalman filter on the smoothing effect of the motion parameter curve, and finally we conducted a comparative analysis based on the objective evaluation parameter PSNR value to obtain the evaluation result.

4.1. Feature points detection comparison

Figure 2 is a comparison diagram of global feature point detection and feature point detection in the set feature dense area. The selected feature dense area is 2/3 below the image in the feature set. It can be seen from the figure that the feature point detection performed in the set feature dense area is compared

\[
\begin{align*}
\dot{x}_k &= A\dot{x}_{k-1} \\
\dot{P}_k &= AP_{k-1}^*A^T + Q \\
K_k &= \dot{P}_k^*H^T(H\dot{P}_k^*H^T + R)^{-1} \\
\dot{x}_k &= \dot{x}_k + K_k(z_k - H\dot{x}_k) \\
\dot{P}_k &= (I - K_kH)\dot{P}_k^*
\end{align*}
\]
with the global detection, the number of feature points detected in the two cases and the effect are not much different.

Table 1. Average operation time for feature points detection

|                    | Global | Feature dense area |
|--------------------|--------|-------------------|
| Average computing  | 46.3   | 36.7              |
| time(ms)           |        |                   |

Table 1 shows the comparison of the average time-consuming situation of feature point detection in the feature dense area and the global of the jitter video frame sequence. It can be found that when the detection effect is roughly the same, the calculation speed of detecting feature points in the feature dense area is higher than detecting the global feature points, which is about 1/6. It can better meet the real-time requirements in the power tunnel inspection process.

4.2. Comparison of smoothing effect of motion parameter curve

The original motion track in Figure 3 contains more jitter components. In the case of sudden changes in the motion state, the improved Kalman filter can perform real-time estimation and judgment, and track the motion trajectory in time, while ensuring the smoothing effect in different states. However, the estimation error of the classic Kalman filter is large, and it is prone to over-cutting.

Figure 4. Improved Kalman filter and classic Kalman filter effect comparison

It can be seen from the effect comparison chart in Figure 4 that the improved image stabilization effect of Kalman filtering can better reduce the over-cutting situation caused by estimation errors. Therefore, the improved Kalman filter in this paper can adapt well to the inspection process of power tunnels where the motion state changes at any time, and ensures the effective real-time smoothing of motion parameters, reducing the loss of information due to over-cutting.

4.3. PSNR value comparison

This paper also uses the objective quality evaluation index Peak Signal to Noise Ratio (PSNR) to verify the effectiveness of the algorithm proposed in this paper. PSNR is judged based on the similarity of images between adjacent frames. The higher the PSNR value, the greater the similarity between frames and the better the image stabilization effect. The calculation formula is as follows:

$$PSNR(S_1, S_0) = 10 \lg \frac{255^2}{MSE(S_1, S_0)}$$  (5)
Among them, $S_1$ and $S_0$ are two adjacent frames before and after, and $MSE$ means calculating the pixel mean square error between the two frames.

**Table 2. Experimental comparison results**

| Algorithm           | Motion state          | Original PSNR (dB) | PSNR After Video Stabilization (dB) | Mean (dB) |
|---------------------|-----------------------|--------------------|-------------------------------------|-----------|
| Original Jittery Video | Fixed scene         | 25.04              | /                                  |           |
|                     | Fast movement        | 17.44              | /                                  |           |
|                     | Movement status change | 22.11             | /                                  |           |
| Kalman Filter       | Fixed scene          | 30.04              | 30.61                               |           |
|                     | Fast movement        | 23.44              | 23.71                               | 27.50     |
|                     | Movement status change | 27.11             | 28.17                               |           |
| Improved Kalman Filter | Fixed scene       | 30.04              | 31.15                               |           |
|                     | Fast movement        | 23.44              | 25.73                               | 28.76     |
|                     | Movement status change | 27.11             | 29.39                               |           |

It can be seen from Table 2 that compared with the traditional Kalman filter, the PSNR value of the algorithm in this paper has different degrees of improvement in the three motion states. The averaged PSNR value in the three states has increased by about 1.26dB. Analysis shows that the image stabilization effect of the algorithm in this paper is better than that of the traditional Kalman filter algorithm.

5. **Conclusion**

This paper proposes a video image stabilization algorithm suitable for power tunnel inspection. The algorithm detects feature points in the selected feature dense area, and it uses the pyramid optical flow method to track the detected feature points, next it combines with the homography matrix to calculate the motion trajectory. Then, it uses the improved Kalman filter to perform real-time filtering processing to obtain the smoothed motion trajectory, and finally obtain a stable video through motion compensation. The experimental results show that the algorithm has better image stabilization effect than the traditional Kalman filter algorithm, reduces the influence of estimation errors caused by the change of the motion state, and increases the calculation speed at the same time, which can better adapt to the real-time needs in the inspection process.

**References**

[1] Rawat, P., & Singhai, J. (2011). Review of motion estimation and video stabilization techniques for hand held mobile video. Signal & Image Processing, 2(2), 159-168.

[2] Matsushita, Y., Ofek, E., Ge, W., Tang, X., & Shum, H. Y. (2006). Full-frame video stabilization with motion inpainting. IEEE Transactions on Pattern Analysis & Machine Intelligence, 28(7), 1150-1163.

[3] Morimoto, C., & Chellappa, R. (1998). Evaluation of image stabilization algorithms. In: 1998 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Seattle, WA, USA. pp. 2789-2792.

[4] Liu, S., Yuan, L., Tan, P., & Sun, J. (2013). Bundled camera paths for video stabilization. Acm Transactions on Graphics, 32(4CD), 1-10.

[5] Grundmann, M., Kvatra, V., Castro, D., & Essa, I. (2012). Calibration-free rolling shutter removal. In: 2012 IEEE International Conference on Computational Photography (ICCP). Seattle, WA. pp. 1-8.