Video Processing (CS046)

Ang Ray Yan

Lim Yao Chong

Introduction / Background Research

Currently, a multitude of professional image manipulation software exists, including Photoshop CS5. However, video manipulation software is receiving insufficient development. With video editing involving the compilation of many manually edited single frames, many users feel frustrated at the tedious work required. Video editing is usually necessary under the circumstance where the lens is obstructed by an object or scratched.

Dust spots stay in the same spot of the frame throughout the video, and thus are moving with respect to the background. This was used by Kang *et. al.* in generating a background image in order to isolate moving objects in PTZ (pan-tilt-zoom) surveillance camera footage. However, their background was generated in accordance with movement data from the camera, which allowed for straightforward transformations of the images to align frames. Normal video footage does not include direct data about the movement of the camera, thus our method aims to recover this movement through homography calculations.

During filming, the camera inevitably moves, rotates or zooms in and out. As a result, frames cannot be perfectly aligned and combined to form large still images such as panoramas, as the frames will be slightly warped with respect to each other. However, how each frame is "warped" compared to another can be recovered in the form of a homography, which describes a geometrical transformation of one projective space (such as the image from a camera) to another (Heckberg, 1989).

Say P_1 in frame 1 corresponds to P_2 in frame 2:

$$\boldsymbol{P_1} = \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} \qquad \boldsymbol{P_2} = \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix}$$

Then:

$$\boldsymbol{P_2}' = \boldsymbol{H}\boldsymbol{P_1}$$

where

$$\boldsymbol{P_2}' = w\boldsymbol{P_2} = \begin{pmatrix} wx_2 \\ wy_2 \\ w \end{pmatrix} \qquad \boldsymbol{H} = \begin{pmatrix} a & b & c \\ d & e & f \\ g & \boldsymbol{h} & i \end{pmatrix}$$

In calculating a homography between 2 images, 4 sets of corresponding points are required.

Objectives

Hence, the objective of this project is to refine the video by replacing obstructions with the obstructed video content in an automated process. The only user inputs required are the original video, the specified starting and ending frames for which the dust spot exists and a general selection window containing the dust spot.

Methodology

Our algorithm for the removal of obstructions such as dust spots from a video involves a two step process. After the user has selected the general area of the dust spot, the necessary homographies required to transform individual frames and align them are first calculated. Using these homographies, a background (sans dust spot) is generated as a template from which the dust spots in each video frame are edited.

1. Homography calculation

Our method first detects corners, defined as intersection of edges, between 2 frames, using the Harris and Stephen corner detection method. Weak corners are detected as well to maximise the number of corresponding points being tested, increasing the accuracy and precision of the homography. Each corner is then characterised by a feature window, which consists of the surrounding 7x7 pixels. The feature windows of each corner are compared between the pictures, and the strongest pairs of corners (above an arbitrary threshold) are kept. However, these pairs may not be true pairs, therefore, as a final step, a RANSAC algorithm is used to calculate the best homography between the two images.

RANSAC (“RANdom SAmple Consensus") is an iterative algorithm which outputs the best model fitting a particular set of data (which may include outliers) by testing potential models obtained using randomly selected data points. In our method, this is implemented by repeatedly taking 4 random pairs and calculating the expected homography between the two frames if these pairs are assumed to be true. This homography is then used to calculate the sum-of-squared-error of non-outlier pairs. The homography which results in a sufficient number of non-outlier pairs (which, based on empirical observation, was at least 80% (see table 2)) and gives the least error was then taken to be the best one.

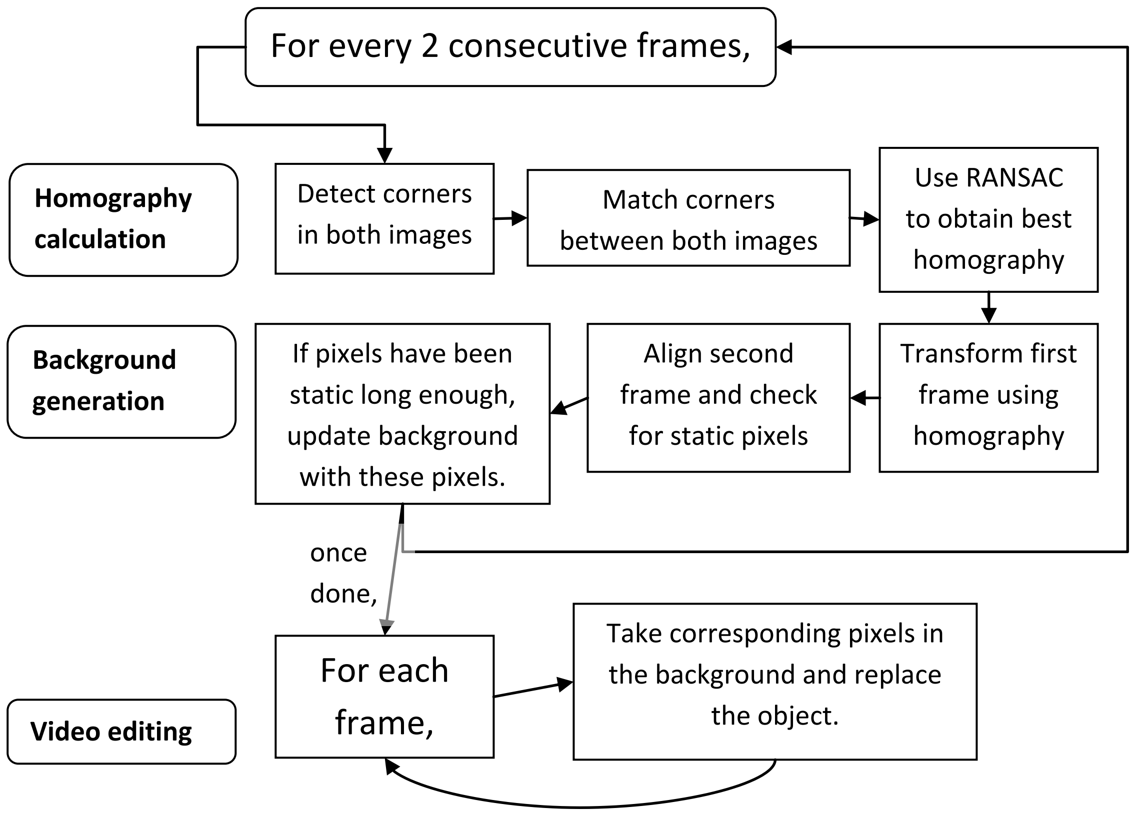
2. Background generation

After the homographies between each consecutive frame (i.e. 1 and 2, 2 and 3, etc.) has been obtained, our method attempts to generate a background image with the dust spot (the “foreground”) removed. This method assumes that every pixel covered by a dust spot is not covered throughout the entire video. As the dust spot would stay in the same spot in the frame while the background continued moving, the dust spot would be moving relative to the background.

Therefore, after transforming all video frames such that they align, our background generation algorithm only takes pixels that have remained unchanged for an arbitrary number of frames (while still taking noise into account) to be the background. The dust spot, which would have moved elsewhere, would therefore be overwritten.

3. Video editing

The resultant background obtained from part 2 would be transformed to align with the last frame being processed. Therefore, when editing the video, the previously obtained homographies are again used to determine which pixels from the background to sample when replacing the relevant dust spot pixels in the original frame.



Results and Analysis

1. Homography calculation

We tested our algorithms on a video of a nature scene, with a large, visible dust spot. Initially, the homographies obtained were inconsistent and had substantial variation each time the program was run. Through observation, we determined that this was due to too few corners being detected, and therefore too few pairs of corners (about 10) for the algorithm to reliably produce an optimal homography (see table). Therefore, we reduced the minimum strength of the corner detected, increasing the number of pairs of corners to around 70, which gave a consistent and more accurate homography, although this also slowed down the program.

"QualityLevel" (corner function parameter - lower value means weaker corners are included)	0.002	0.003	0.004	0.005	0.01
Average potential pairs of corners obtained	79	72	57	47	27

Table 1: Number of pairs obtained

"QualityLevel" (corner function parameter)	0.01	0.005	0.001
Time taken to obtain homographies between 10 frames (seconds)	21.78	39.93	77.56

Table 2: Time taken to obtain homographies

After processing the potential pairs of corners through the RANSAC algorithm, an average of 97% of potential pairs of corners were true pairs (see table below).

Mean	Median	Max	Min
0.957724	0.972973	1	0.826087

Table 3: Ratio of true pairs of corners to potential corner pairs ("QualityLevel" = 0.001)

2. Background generation

For background generation, we experimented with various thresholds for considering a pixel to be "unchanged". Higher thresholds led to more pixels being replaced in fewer frames, as expected, however, this had the side effects of replacing unwanted pixels, making the background look "noisy", as well as reintroducing the dust spot. As a result, we found that setting a threshold of 0 was best as this would affect the edited video quality the least, but still replace the dust spot, albeit after more frames.

Generated background, using 30 frames

3. Video editing

Before and after - first frame of video

Limitations and Extensions

1. Runtime

This method, while accurate, is also slow. However, our results show that the runtime increases linearly, with one frame taking an average of 10 seconds (see graph). This is mostly due to the comparison of the feature windows characterising each corner.

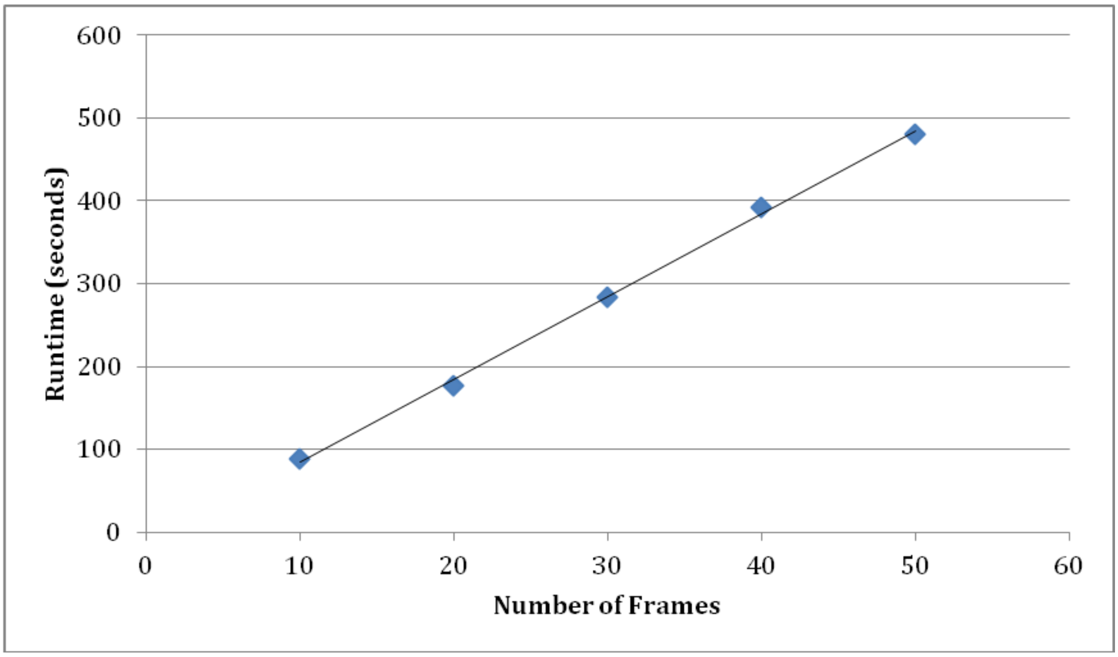


Table 4: Runtime

2. Dust spots that cannot be removed

Under certain circumstances, the camera may not move sufficiently such that every single obstructed content will be revealed in any other frame. Without real information, it is impossible for any algorithm to retrieve lost data. Hence, a possible extension includes the combination of content-aware algorithms to generate the content for replacement instead.

Bibliography of References

1. Kang, S., Paik, J., Koschan, A., Abidi, B., & Abidi, M. A. (2003). *Real-time video tracking using PTZ cameras*. In *Proc. of SPIE 6th International Conference on Quality Control by Artificial Vision, Vol. 5132* (pp. 103-111). Retrieved from http://imaging.utk.edu/publications/papers/2003/kang_qcav03.pdf
2. Heckbert, P. S. (1989). Projective Mappings. *Fundamentals of Texture Mapping and Image Warping* (pp. 17-21). Berkeley: CS Division, U.C. Berkeley.
3. Stegmann, M. B. (2001). *Image warping*. Unpublished manuscript, Informatics and Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark.
4. Ostiak, P. (2006, April). In L. Szirmay-Kalos (Chair).*Implementation of HDR panorama stitching algorithm*. Paper presented at Central european seminar on computer graphics, Bratislava, Slovakia. Retrieved from http://www.cescg.org/CESCG-2006/papers/Szczecin-Ostiak-Piotr.pdf
5. Steedly, D., Pal, C., & Szeliski, R. (2005). *Efficiently registering video into panoramic mosaics*. Paper presented at International conference on computer vision, Beijing, China. Retrieved from http://research.microsoft.com/pubs/75621/Steedly-ICCV05.pdf