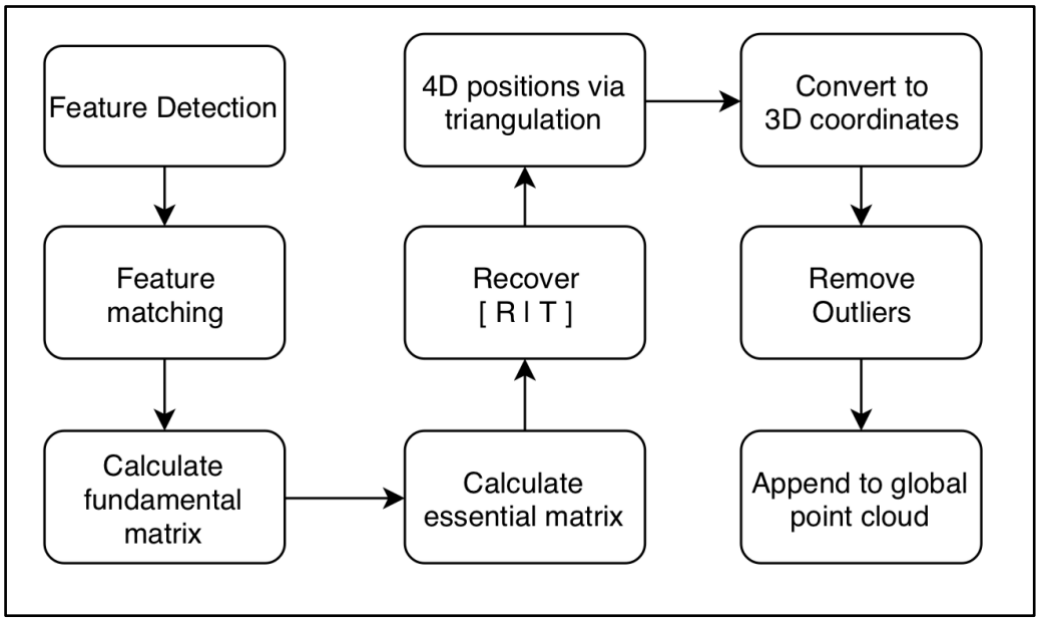
Structure from Motion: creating a 3D world from an image sequence

Pipeline



Approach to Problem

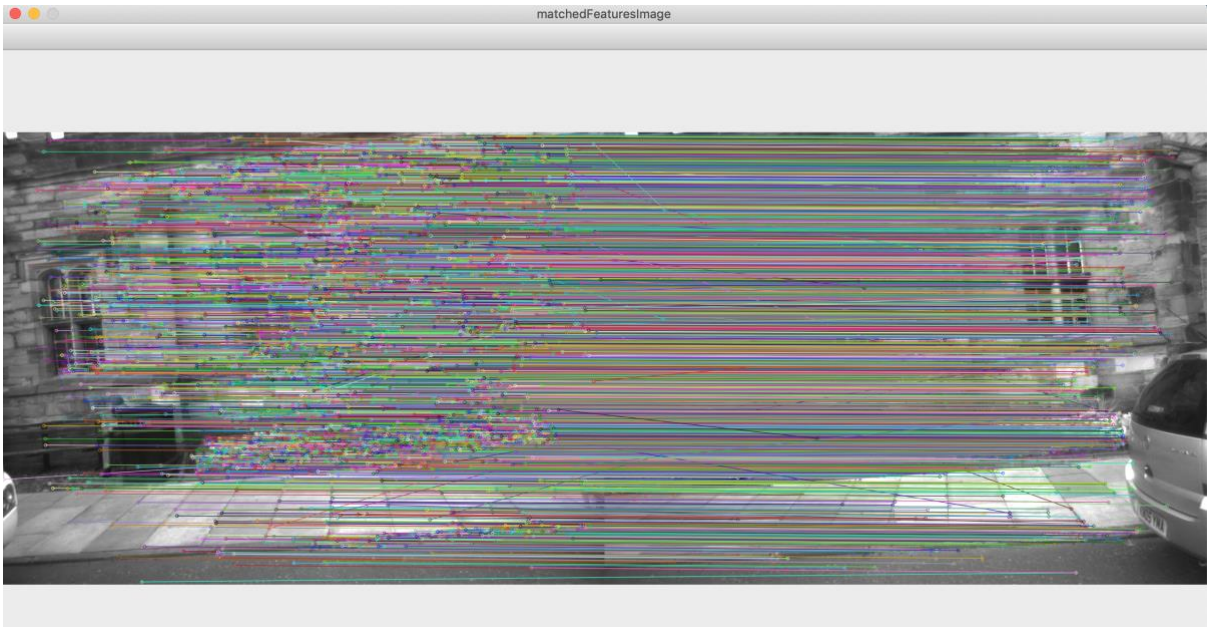
NB. The algorithm works on image/frame pairs: *frame* t and *frame* t+1 at time t+1.

The structure from motion pipeline begins with feature detection and extraction with the use of the SURF operator which computes the descriptor of the detected feature points for both images as illustrated in Figure 1. A Hessian threshold of 350 is used.

Figure 1- *Detected Feature points*

From the detected feature points in both images, the algorithm conducts sparse feature descriptor matching. For this, a FLANN based matcher is used which is faster than brute-force matcher. The matches are then filtered using Lowe's ratio test to get the best match. This feature matching between a pair of images is illustrated in Figure 2.

Figure 2- *Sequential feature matching between a pair of images*



From the corresponding feature matches between a given pair of images, the fundamental matrix is calculated through the use RANSAC. For this we set the maximum distance from a point to an epipolar line, beyond which the point is considered an outlier, to 0.5 and the level of confidence to 99%. These strict parameters help us to achieve the best approximation of F. From this, we compute the essential matrix through the equation:

$$E = K^T F K$$

where K is the intrinsic camera parameters. An example output of these matrices are shown in Figure 3.

Figure 3- *Fundamental and Essential Matrix*

```
Fundamental Matrix:
[[-3.04931861e-19 -1.42681006e-16  3.13117587e-14]
 [ 1.43982049e-16  2.15485182e-18 -2.00000000e-01]
 [-3.14470672e-14  2.00000000e-01  1.00000000e+00]]


Essential Matrix:
[[-1.13641092e-12 -3.26793066e-10 -5.92317783e-11]
 [ 3.29772942e-10  3.03317982e-12 -2.37285046e+02]
 [ 5.93349363e-11  2.37285046e+02  1.00000000e+00]]
```

We then iterate over all point correspondences used in the estimation of the fundamental matrix, using the F-mask to normalise and homogenise the image coordinates, giving us a set of inliers and removes outliers.

From the essential matrix, the camera-to-camera image transformations is recovered via singular value decomposition (SVD) of the essential matrix. This decomposes E into rotational and translational matrices [R | t]. This decomposition has four possible solutions and only one of them is the valid camera matrix. To get the correct one we find the one that predicts that all the imaged key points lie in front of both cameras. An example output of these matrices are shown in Figure 4.

Figure 4- *Rotation and translation matrices*

```
R:
[[ 1.00000000e+00 -9.19430933e-15  1.30750365e-14]
 [ 9.16673765e-15  9.99997780e-01  2.10716562e-03]
 [-1.30943814e-14 -2.10716562e-03  9.99997780e-01]]


T:
[[ 1.00000000e+00]
 [-2.43818825e-13]
 [ 1.37721728e-12]]
```

As we know the car cannot move in the z-plane, we repeat the above steps of calculating [R|T] until we have a translation vector with a small z-component.

From the rotation matrix, we construct the 3D real-world coordinates of the image points by making use of epipolar geometry as shown in Figure 5.

Figure 5- *Triangulation Theory*

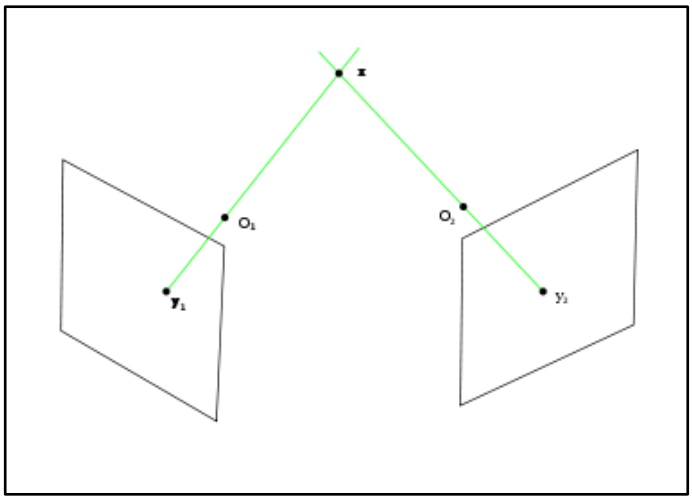
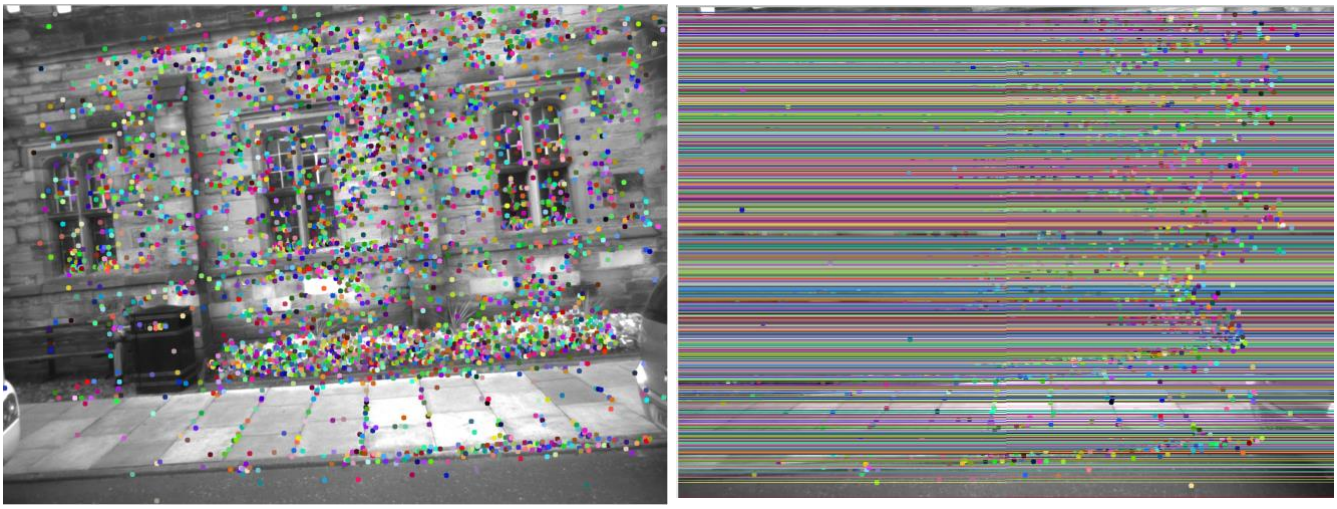


Image taken from: *https://en.wikipedia.org/wiki/Triangulation_(computer_vision)*
The resulting set of 3D points contains outliers and so to remove these, the average position of all the 3D points is calculated and then points which are further away than a given threshold value are removed.

Images are then rectified using the intrinsic camera matrix K, E and the projection matrix as well as the camera distortion coefficients d. Epipolar lines were plotted which were horizontal, as shown in Figure 6, meaning that the camera had been fully rectified.

Figure 6- *Image Rectification*



In order to perform dense structure recovery, a disparity map was created through the use of stereo correspondence using the semi-global block matching algorithm, as shown in Figure 7.

Figure 7- *Disparity Map*

The disparity map is re-projected back to 3D points through the use of the disparity-to-depth mapping matrix Q recovered during image rectification.

Evaluation (Quantitatively measuring performance)
Sequence Tested on: *OH-sfm-test-sequence-DURHAM-2015-03-27-calibrated*

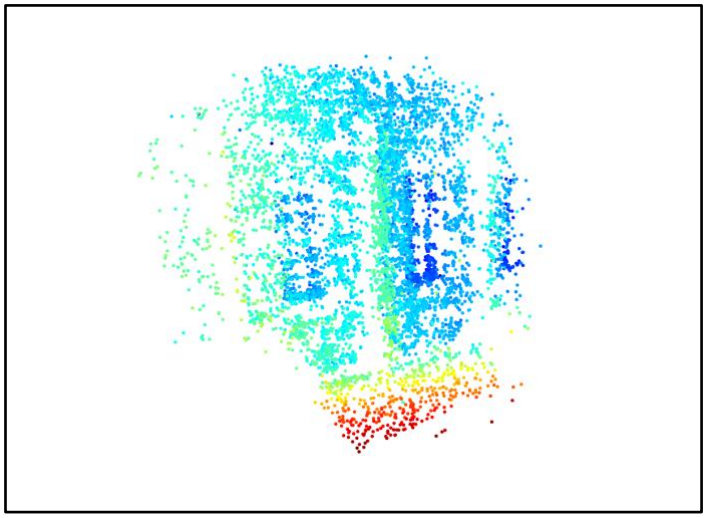
Based on a representative subset of 20 sample images, the approach was evaluated by considering the success of the solution at various stages throughout the pipeline considering different metrics.

For speed, we measure how long it takes the whole pipeline.
For quality of output, three external persons evaluated the quality of the structure-from-motion output for each image pair, which was then averaged and rounded to the nearest integer.

Frame Number	Number of Feature Matches	Time for pipeline (s)	Quality of sfm output /5
2	3881	3.25	4
3	3754	3.14	4
4	3579	2.77	4
5	3487	2.48	4
6	3231	2.63	4
7	3033	2.23	3
8	2708	2.63	3
9	2094	1.35	4
10	2070	1.39	3
11	2147	1.47	2
12	2163	1.54	3
13	2288	2.01	3
14	2346	1.65	2
15	2535	1.68	2
16	2661	1.68	3
17	2792	1.72	3
18	2875	1.74	3
19	2999	1.81	3
20	3094	1.86	3

Average time for processing pipeline: 2.24s.
Average number of matched feature points: 2686

Evaluation of algorithm success
Example 1

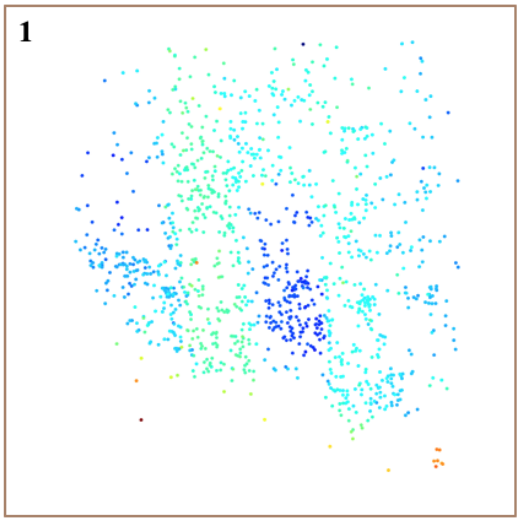


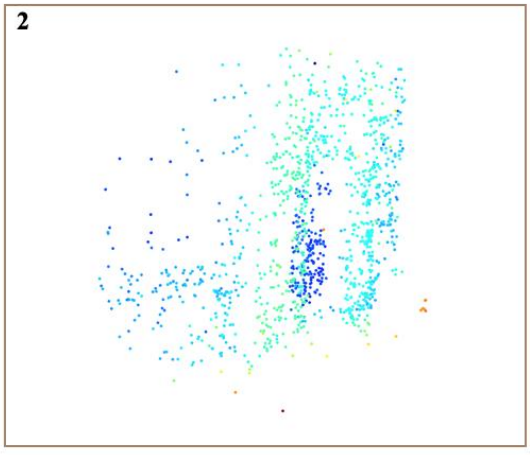
SFM pipeline successfully determines depth of image scene to a great degree success.

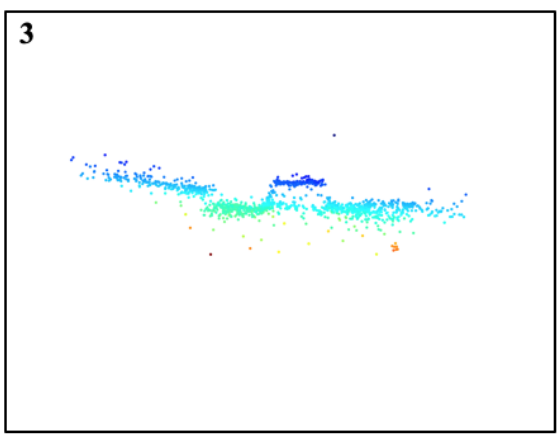
Link to YouTube video analysing the scene above:
https://www.youtube.com/watch?v=gMZziaVk04I

Example 2

The images of the 3D structure below are from in front of the door (1), from a left view (2) and one from above (3).



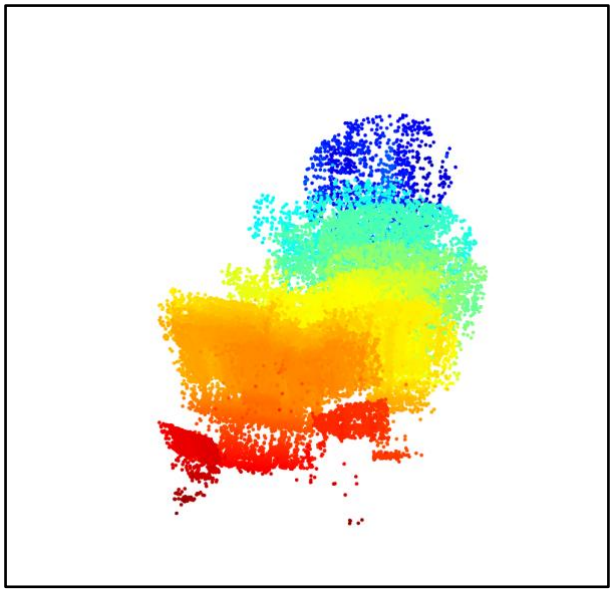




Example 3

Doesn't pick out the car due to a lack of feature points. Ignores moving scene objects such as people.

Figure 7- Final Structure from motion scene reconstruction



Link to YouTube video of 3D realisation of above scene:

https://youtu.be/wlFyWJUIT4E

Conclusion

This sfm approach works with varying degree of success, achieving good results for individual pairs of frames however the "stitching" is limited as shown Figure 7.

References

- Michael Beyeler - OpenCV with Python Blueprints: Packt Publishing, ISBN 978-178528269
- Toby Breckon- Durham University