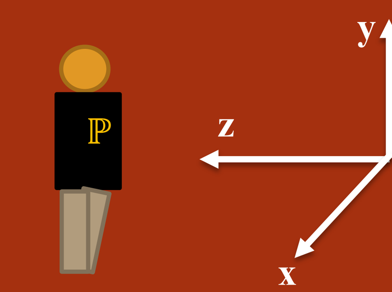
A Graph-based Algorithm to Video Stitching

Jiatong Yu

COS429 Computer Vision, Princeton University



MOTIVATION

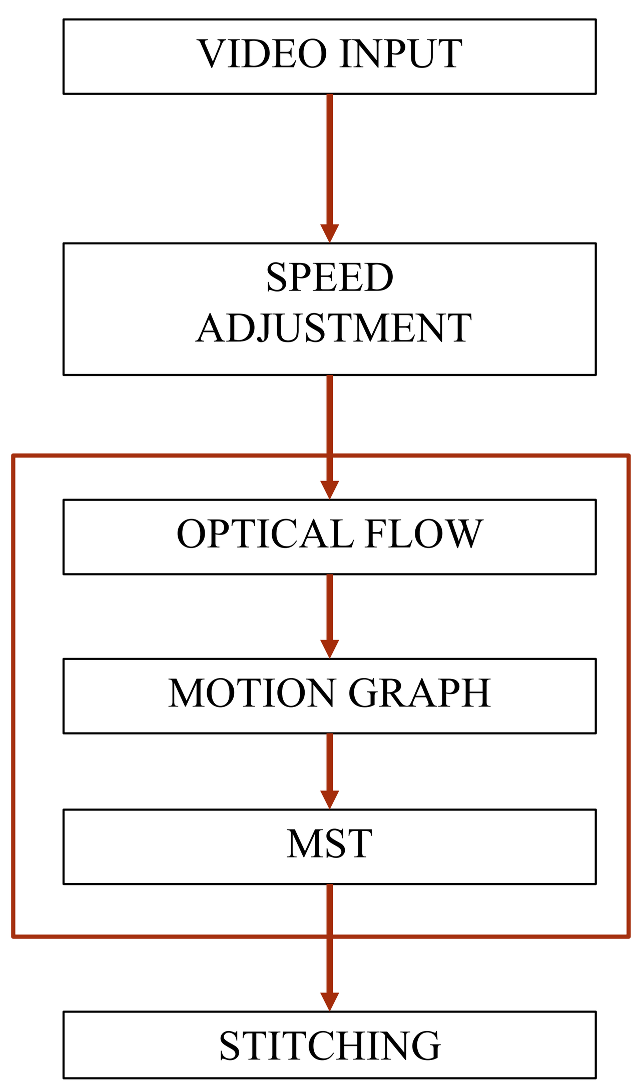
- Portable device panorama allows people to capture their favorite scenes and moments on a single canvas.
- Taking a collection of pictures for panorama is both harder to manage and more error-prone. Video input is more convenient and robust to client-side risks, such as photos too distinct to perform panorama.
- The majority of current video stitching models either 1) relies heavily on hardware and camera position, or 2) overlook the advantage of structural similarity between adjacent frames.

RELATED WORK

Brown and Lowe[1] developed a multiple-image stitching algorithm that uses bundle adjustment to avoid cumulative distortion and blur, unlike a baseline model that iteratively joins adjacent images that does not guarantee structural similarity. Conceptually speaking, the bundle adjustor picks the next best match of existing bundle, calibrate its perspective, and stitch it into the bundle.

However, with video inputs, we can **develop a more efficient bundle adjustor** (compared to the proposed non-linear least square minimization) **by tracking camera movement**. In particular, we observe that frames with similar camera position should inherintly be each other's best match. Therefore, we set out to reconstruct the camera movement in a graph-based algorithm.

ALGORITHM OVERVIEW



Speed Adjustment

Users may not rotate their camera at a uniform speed, therefore the first challenge is to selecting frames that **accommodate varying camera moving speed**. We propose an algorithm such that the probability of selecting one frame is proportional to the camera moving speed at that frame.

Graph-based stitching

The algorithm uses optical flow to calculate estimated camera movement between each adjacent selected frame. An undirected graph is constructed to reflect such camera movement, in which the edge weight reflects the Euclidean distance of x- and y-axis camera movement between two frames. Minimum Spanning Tree (MST) algorithm thus **produces a stitching order that conceptually reflect the bundle adjustment method**[1].

SPEED ADJUSTMENT

The speed adjustment algorithm treats the selection of each frame as a **Bernoulli Trial** with

$$P(X_i = 1) = c\,T_i,$$

And set

$$E[\Sigma X_i] = N$$

Where $X_i = 1$ means select frame, T_i is the estimated camera moving speed at that frame, and N is the number of frames we want to extract.

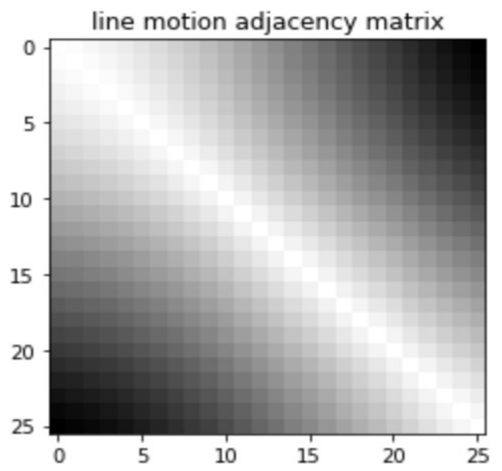
Solving for c gives the speed-adjusted selection. Running the algorithm on a speed-variant video produces a reasonable panorama, while uniform frame extraction fails to stitch many frames.

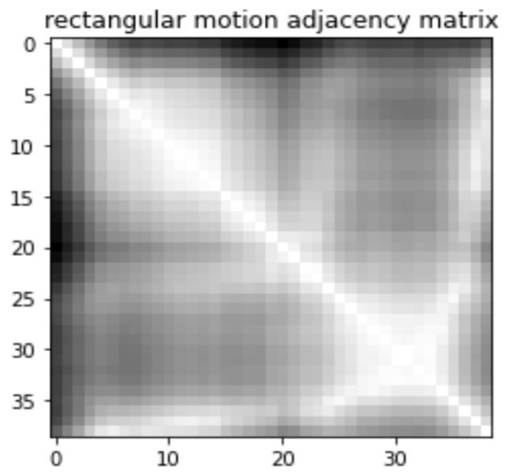
Results on Same Video Input

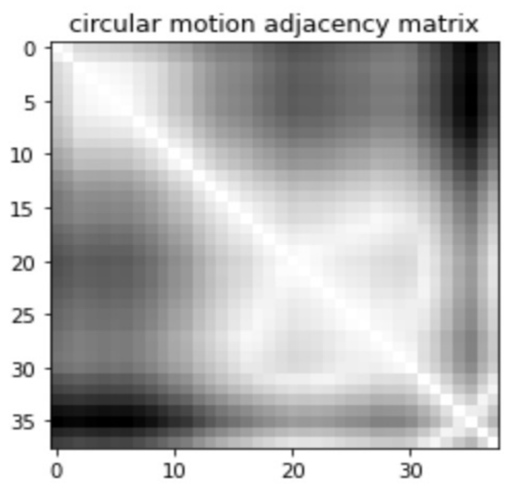
Graph Panorama without Speed Adjustment
(Failed to stitch frames to the right.)

Graph Panorama with Speed Adjustment

GRAPH-BASED STITCHING







Optical Flow

- Optical Flow produces a 2D vector that describe keypoiny movement between two images
- Assuming still scene, keypoint movement suggests camera motion

Motion Graph

- We calculate x-axis displacement by

$$\mathbb{D}_x(m,n) = \sum_{i=m}^{n-1} \Delta x_{i,i+1}$$

and same for y-axis, assuming m > n WLOG

- Graph node is each frame, and edge is the Euclidean distance using previous equation
- The graph is tested with three typical camera movements:
  1) Horizonta: camera does not move up or down
  2) Rectangular: move right, move up, move left
  3) Circular: camera rotates around z-axis
- As shown in the left, the **adjacency matrix reflect information about the actual camera movement**.

Minimum Spanning Tree

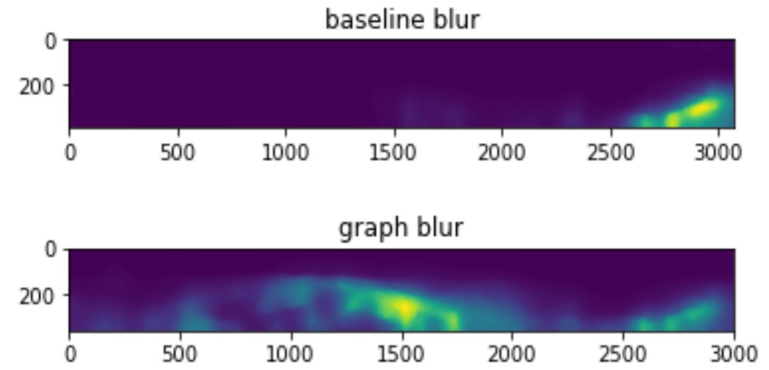
- As each edge reflect the estimation of camera distance from two node frames, the MST algorithm finds a sequence to stitch all selected frames so that frames are stitched with most similar "best matches"

EVALUATION

SAMPLE RESULT

Baseline Result *(Video downloaded from a github repo)*

Graph-based Result




```
baseline blur score:  23.00%
graph blur score:  13.02%
```

METRICS

We adopt blurriness as an estimation of the effect of graph-based algorithm because, as Brown and Lowe suggested, efficient bundle adjustment should remove blurriness and distortion.

DISCUSSIONS

- Testified that video stitching could leverage the sequential content similarity between frames to produce better results and potentially more efficient algorithms
- Graph-based algorithms can implement the concept of bundle adjustment with lower complexity.
- Exist alternatives to optical flow that could be explored. Improvement could be made to calculate more accurate and apprehensive camera movements
- Exist alternatives to MST calculation. One immediate possibility is max-flow/min-cut algorithm that partition the graph into most structurally similar regions.

REFERENCES

[1] M. Brown and D. G. Lowe. Automatic panoramic image stitching using invariant features. *Int. J. Comput. Vis.*, 74(1), 2007.

[2] S. A. Golestaneh and L. J. Karam. Spatially-varying blur detection based on multiscale fused and sorted transform coefficients of gradient magnitudes. *Computer Vision and Pattern Recognition*, 2017.

[3] J. Kopf, M. F. Cohen, and R. Szeliski. First-person hyper-lapse videos. *ACM Trans. Graph.*, 33(4):78 – 88, 2014.

[4] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.*, 60(2):91–110, 2004.

[5] Z. Wang, A. C.Bovik, and A.R.Sheikh. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[6] Z. Wang, E. P. Simoncelli, and A. C. Bovik. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[7] M. Zheng, X. Chen, and L. Guo. Stitching video from webcams. *Proceedings of the 4th International Symposium on Advances in Visual Computing*, pages 420 – 429.