Fast feature and accurate point registration by bigraph oriented edges voting

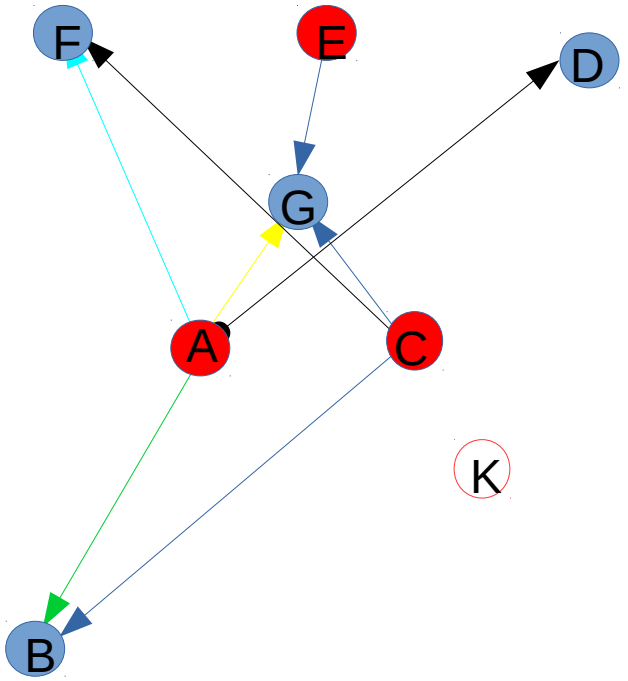
Samsonov Dima 14 August 2017

- Current pairing(registration) schemes have several problems:
- a) Lack of precise orientation and scale
- b) slow O(N^2) pairing feature search
- c) don’t use native local connectivity for small local affine deformation

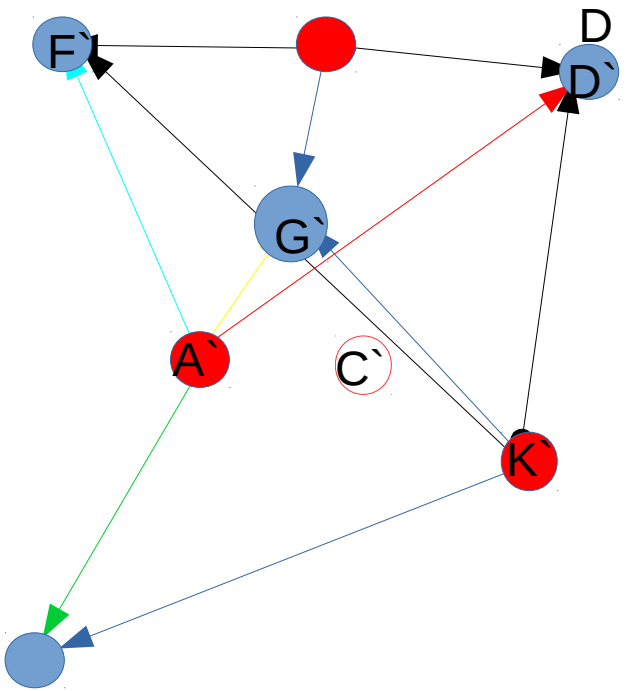
Let try to develop pairing algorithm without these cons

Find vertices and oriented edges

Left image

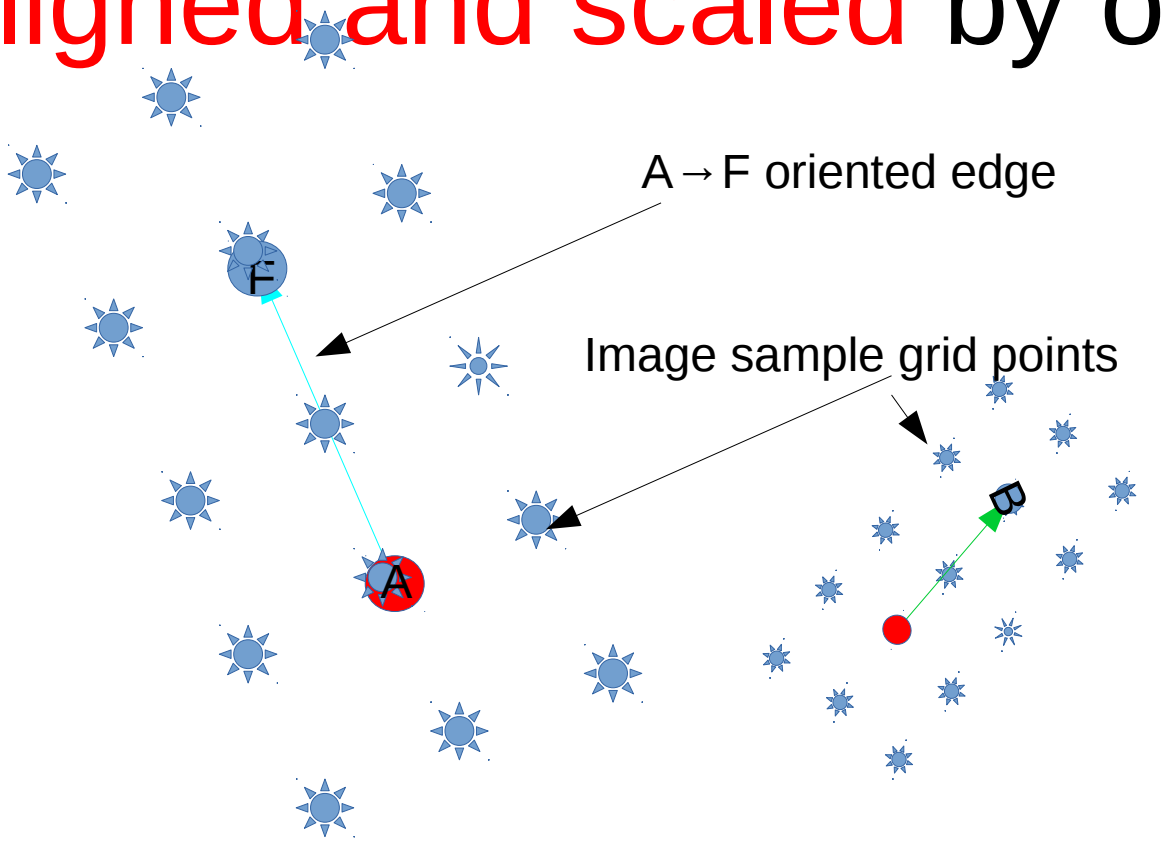


Right image



1 Find featured points,blob (vertices) on each image (like local max DoG,LoG,DoH...etc)
2 Set vertex color as “red” if DiffOfGaussians is positive or otherwise “blue”
3 Connect each red vertex with ~6-10 closest blue vertices.
So we have array of ~N/2(red) + ~N/2(blue) verices and array ~(N/2*10) of oriented edges

Calculate features and hash on grid aligned and scaled by oriented edge



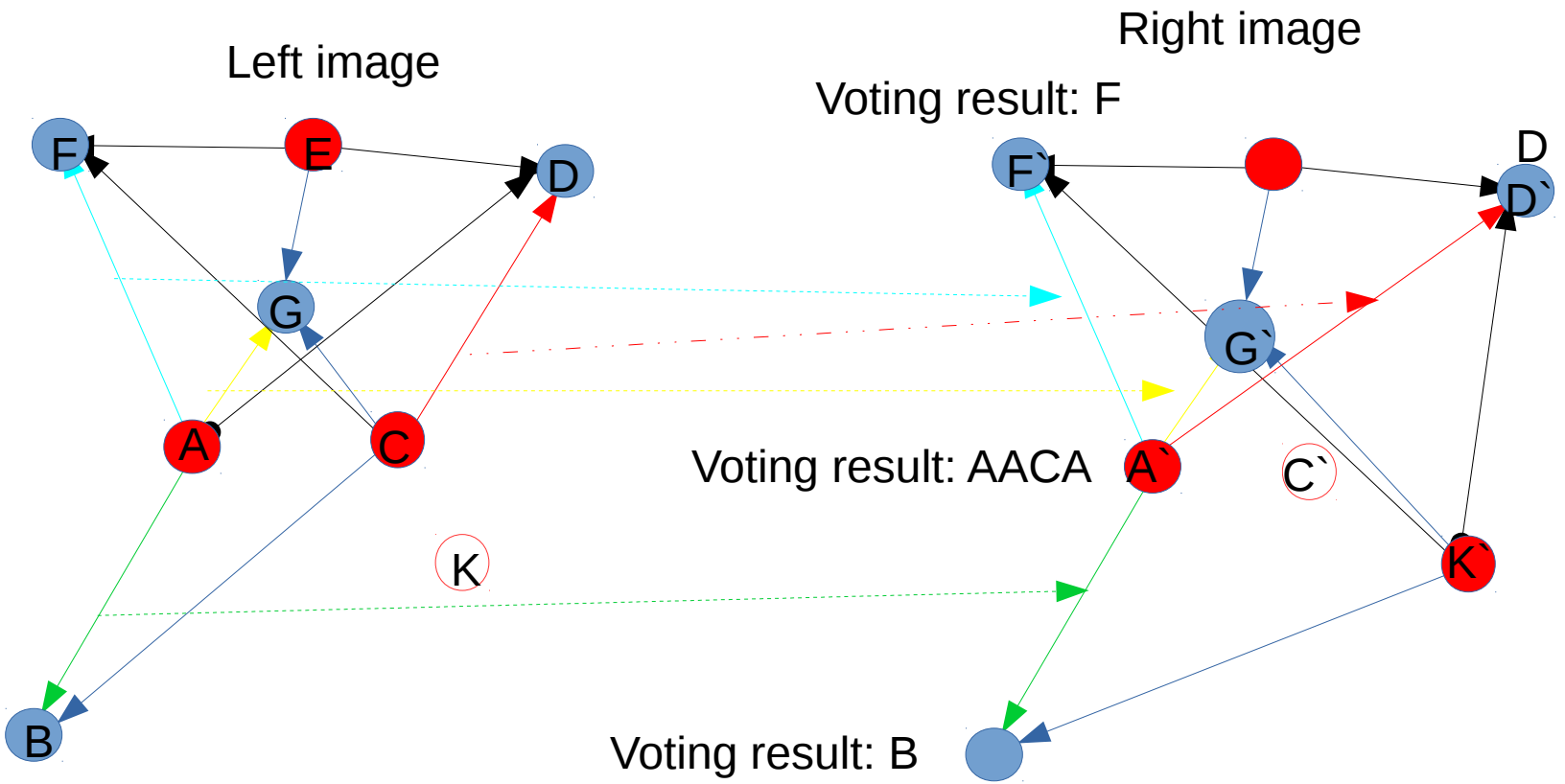
Hash value 0b11100...: vertices A F

Hash value 0b01001...: vertices G B

4 Sample (bilinear) smoothed image around each edge and calculate ~24bit hash value as sign of samples difference.

5 Reduce ~24 bit hash value to appropriate bit length through some hash16-hash12 crc function for fast O(1) LUT search

Edges voting vertices pairing



Hash value 0b11100: vertices A F

Hash value 0b01100: vertices A G

Hash value 0b01100: vertices C D

Hash value 0b01110: vertices A B

Hash value 0b11100: vertices A` F`

Hash value 0b01100: vertices A` G`

Hash value 0b01100: vertices A` D`

Hash value 0b01110: vertices A` B`

6 For each edge from left image put to vertices of edge(s) with same hash on right image indices of edge vertices
So we have point A` from the Right image paired to Point A from the Left image by voting 3 times for A and one time for C

Example of image registration without outliers filtering

Example of image registration without outliers filtering fullHD 40 fps

Pros and Cons

Pros:

- Rotation and Shift invariant
- Simple code
- Mostly no outliers (due voting scheme)
- Fast pairing (registration) suitable for real time FullHD 30FPS video ,with about 2000-10000 (6000 paired in 6 ms) for vertices per frame on modern dual core cpu or 640x480 30 fps video on rasberrypi3.

Cons:

- Relative sense to scale (but can be modified to work with pyramid)
- Sense to significant projective distortion

Applications

- SLAM and 3d video reconstruction
- Super resolution(due sub-pixel precision)
- Video De-noising
- Video compression
- Image recognition
- Face recognition
- 2d and 3d x-ray registration

Link to open source C++ implementation

- Complete source code in https://github.com/sdima1357/bigraph_image_registration_demo

Links

- SIFT https://en.wikipedia.org/wiki/Scale-invariant_feature_transform
- SURF https://en.wikipedia.org/wiki/Speeded_up_robust_features
- SLAM https://en.wikipedia.org/wiki/Simultaneous_localization_and_mapping
-
-