COMPUTER SCIENCE
& ENGINEERING
TEXAS A&M UNIVERSITY

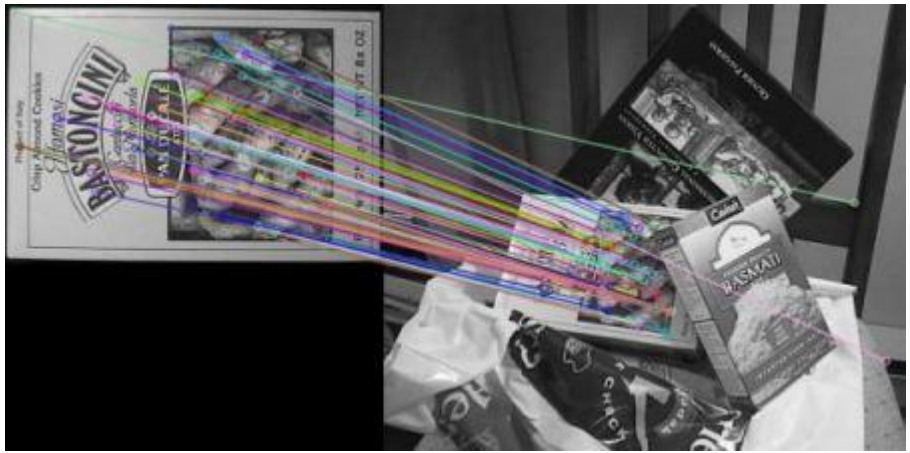
Dimension Reduction for SIFT descriptors to improve matching efficiency

Jicheng Gong

11.30.2017

SIFT (Scale-Invariant Feature Transform)

- Scale space peak selection
- Key point localization
- Orientation Assignment
- Key point descriptor



Key point descriptor

- Create a set of orientation histograms on 4x4 pixel neighborhoods with 8 bins each.
- Histograms are computed from magnitude and orientation values of samples in a 16 x 16 region around the key point.

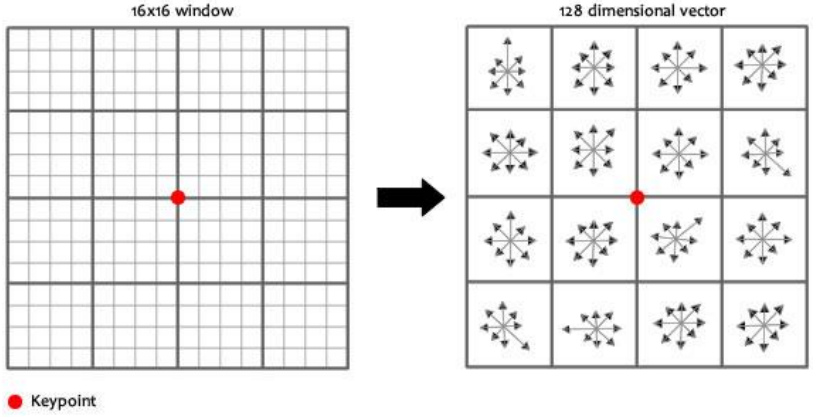
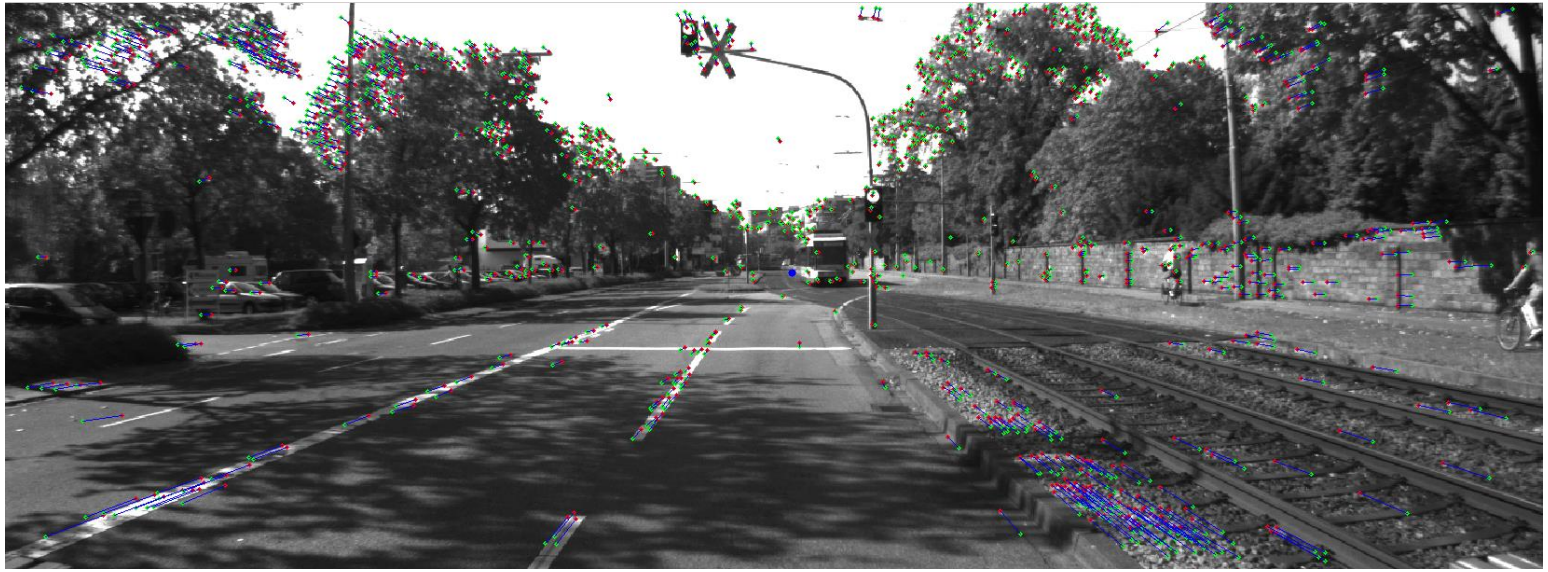


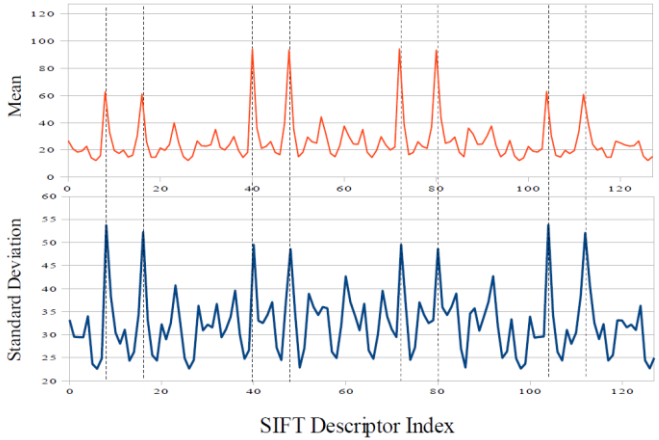
Image Matching

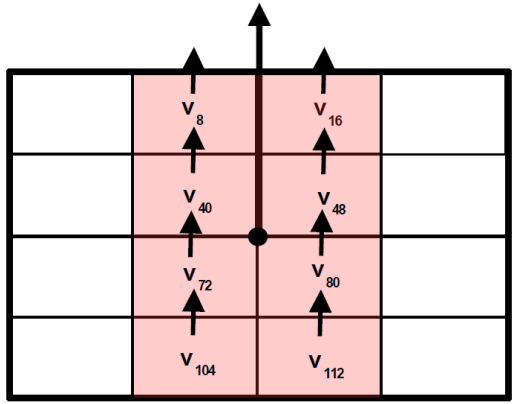


SIFT-HHM

ENGINEERING
TEXAS A&M UNIVERSITY

- 8 mean and standard deviation peaks
- Corresponding descriptor vectors are in line with key point's orientation





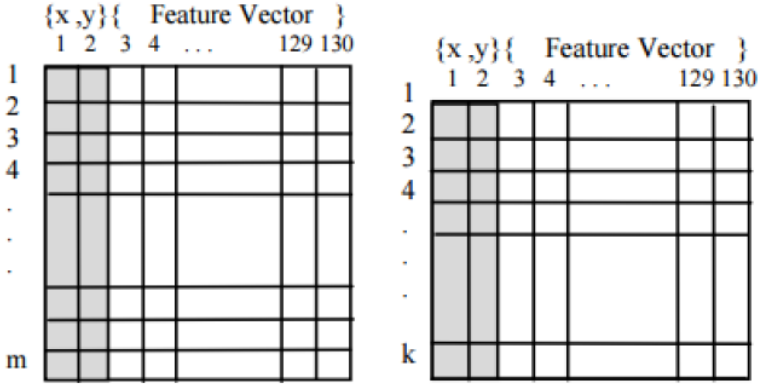
Euclidean Distance Vs Inner Product

- The original matching method in SIFT is calculating the smallest and second smallest Euclidean distance between two vectors. If the distance_1/distance_2 is smaller than a threshold such as 0.8 or 0.65, we will mark the pair as a good match pair.

$$L_o = \sqrt{\sum_{i=1}^{n} \left(x_i - y_i\right)^2}$$

$$P = x_1 * x_1' + x_2 * x_2' + x_3 * x_3' + \ldots + x_{128} * x_{128}'$$

SIFT



$$P = M \cdot K^T$$

$$Recall = \frac{Correctmatchesretrieved}{Totalnumberofcorrectmatches}$$

$$1 - Precision = \frac{Incorrectmatchesretrieved}{Totalofmatchesretrieved}$$

Feature Selection

ENGINEERING
TEXAS A&M UNIVERSITY

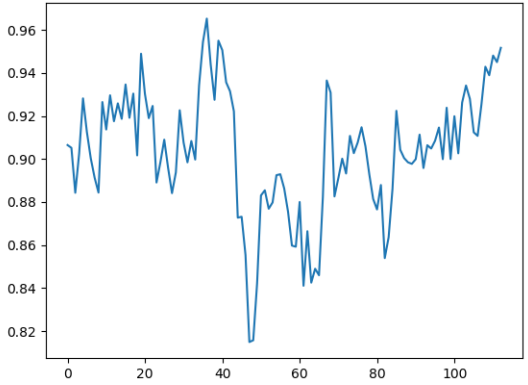
Unsupervised

- Variance Threshold
- PCA

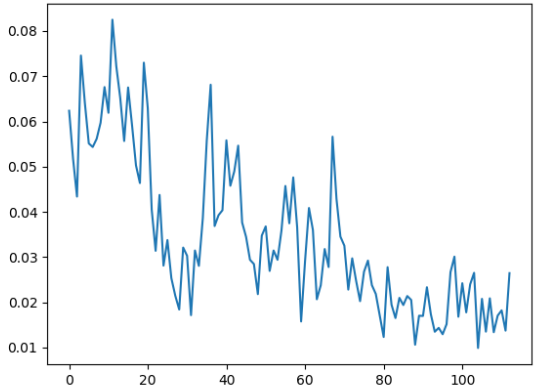
Supervised

- Random Forest
- LASSO with L1 and L2 norm

Results

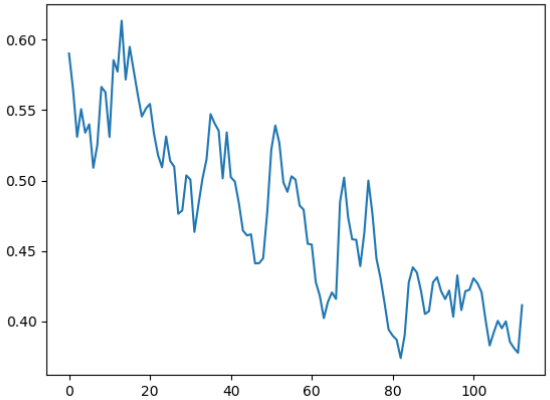


Recall Rate for Inner Product method

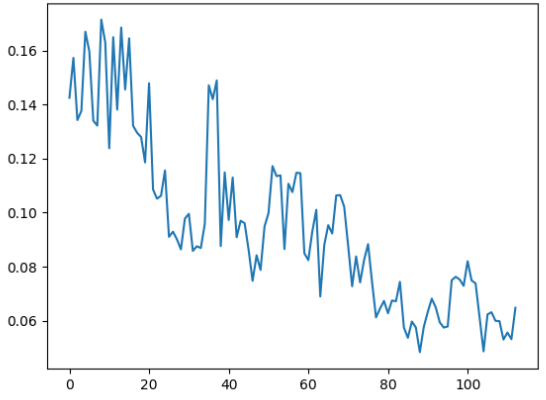


Error Rate for Inner Product method

Results



Recall Rate for PCA-64 method



Error Rate for PCA-64 method

Results

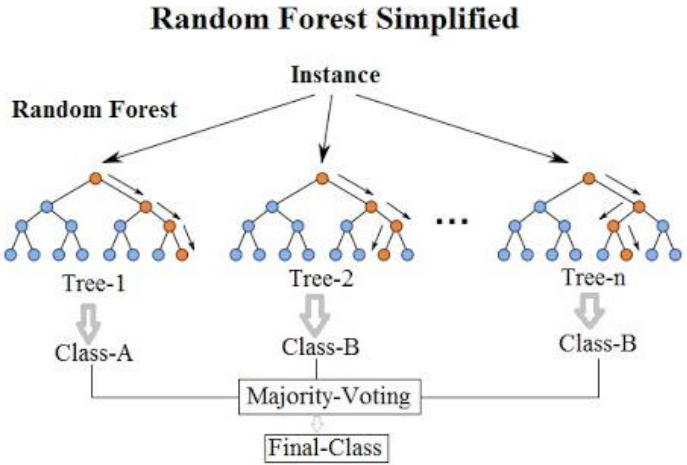
ENGINEERING
TEXAS A&M UNIVERSITY

	Recall	Error Rate
Inner Product	90%	5%
VAR-80	50%	7%
VAR-64	40%	10%
VAR-32	22%	12%
VAR-16	5%	3%

	Recall	Error Rate
Inner Product	90%	5%
PCA-80	25%	5%
PCA-64	50%	10%
PCA-32	25%	10%
PCA-16	8%	5%

Random Forest

$$prediction = bias + feature_1 contribution + \ldots + feature_n contribution.$$



COMPUTER SCIENCE
& ENGINEERING
TEXAS A&M UNIVERSITY

Thank You!

Jicheng