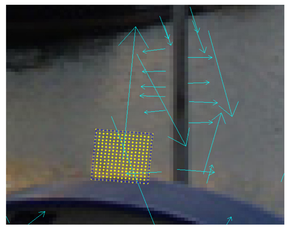
Object Recognition

Using SIFT & a Vocabulary Tree

Visiondays - Wednesday 28th of May - 2008

Figur 1: Image keypoints

Figur 2: All Color-Patches



Figur 3: A single Color-Patch

Abstract

This project focuses on the method of fast object recognition using SIFT-descriptors and a Vocabulary tree. The objects of interest are known tourist sites found in Copenhagen. The use of color information, PCA and Multiple view Geometry are applied to improve the success of the recognition.

Application idea & Database construction

The application idea of this project deals with the scenario of a tourist visiting Copenhagen with his camera and cell phone. The tourist uses his cell phone to gain information about the specific site by sending a photo to an online service, which will recognize the object in the picture and send relevant information back. Another possibility is to use the system to organize the photos automatically. The tourist uploads pictures to his PC and use the application software to automatically arrange the pictures according to location and collect online information of each site.

The database used for recognition has to be constructed of several images of the same location taken from different angles. For now the database consist of 97 images of 17 tourist sites in Copenhagen.

SIFT

SIFT (*Scale invariant feature transform*) is an efficient tool for finding distinctive image features in scale-space. Each feature is represented as a 128 dimensional descriptor vector. In figure 1 the descriptors are shown as keypoints in the image with a given scale and direction.

Building a Vocabulary Tree

The SIFT descriptors found for all images in the database are to be hierarchically quantized into a *vocabulary tree* for fast recognition. The steps of this representation are:

- The features are clustered using *k-means* clustering based on user given values of tree levels and branch factor.
- Each leaf node represents a *visual word* in the vocabulary and using *inverted files*, these words are associated to the images, where they appear.

The vocabulary tree represents the *bag-of-words* model, which is a way of categorizing images into representatives. The visual words of each image is collected into a frequency vector, which is weighted according to the distribution of visual words. The query image is then propagated down the tree to find the nearest leaf node, and a comparison up against the frequency vectors are made using the *L1* distance norm.

Adding Color information

The SIFT descriptors are only found in the gray-scale representation of the image, so no color information are actually incorporated in the recognition. The color information is added using box-like patches around each SIFT-keypoint. The size and orientation are varied according to the keypoint scale and orientation. The mean pixel-values of the patches for each color-channel (RGB) is then added to the descriptor vector.

PCA (*Principal components analysis*) is used afterwards to reduce the dimensionality of the feature vector to around 10 to 20 dimensions.

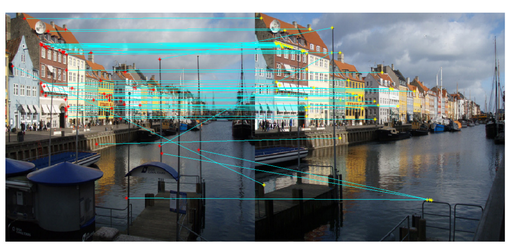
Multiple view Geometry

The geometric relation between the query image and the retrieved images can be examined to remove some of the false negative results. This is computational expensive, so it is only made to the top 10 recognized images. The images are compared using the visual words of the images to find a matching between keypoints. Based on these corresponding points, the fundamental matrix is estimated using *RANSAC*. The matching is then reevaluated for each keypoint using *epipolar geometry*.

Results

The results of the recognition for a query image of *Nyhavn* is shown. In figure 2-3 the color-patches is shown for a single and all keypoints. In figures 4-7 the visual word matching, the epipolar geometry match and the final matching are shown. The final result before and after the use of multiple view geometry is shown in figure 8. Notice the rearrangement of some of the top ranked images.

Author: *Peter René Bolvig Stentebjerg*

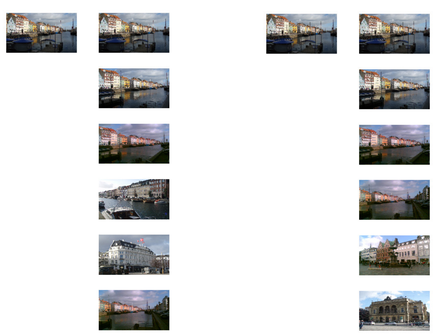


Figur 4: Matching using visual words

Figur 5: Point in query image

Figur 6: The epipolar line. The blue point is the corresponding one.

Figur 7: Final matching after using epipolar geometry



Figur 8: Top 6 recognized images before (left) and after (right) using Multiple view Geometry.

Informatics and Mathematical Modeling