Simultaneous Recognition and Homography Extraction of Local Patches with a Simple Linear Classifier

Stefan Hinterstoißer[1], Selim Benhimane[1], Vincent Lepetit[2], Pascal Fua[2], Nassir Navab[1]

[1] Computer Aided Medical Procedures (CAMP), Technische Universität München, Germany
[2] Computer Vision Laboratory (CVLAB), École Polytechnique Fédérale de Lausanne, Switzerland

Overview

Problem

·Previous approaches (Leopar[1] - [Same authors,CVPR'08]) showed that we can efficiently estimate the 3D pose of a poorly textured object by learning the patch appearance. Leopar[1] is performed in 4 steps:

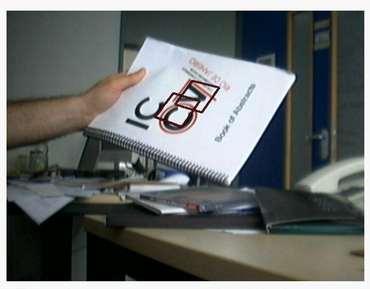
(a) Pre-classify feature points with e.g. Ferns[2]
(b) Rough orientation estimation of the patch with respect to the feature point identity
(c) Rectification refinement by applying a Template Matching algorithm[3]
(d) Outlier removal by simple correlation measure

Leopar[1] gives much better results concerning the accuracy and repeatability of the pose than affine region detectors but still suffers from a decreasing robustness towards large viewpoint changes. Reasons for that:

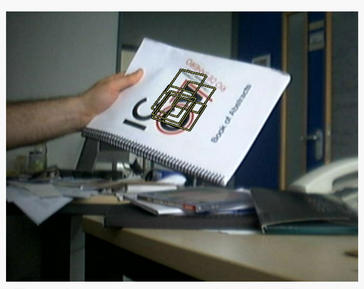
- Error-prone pre-classification of initial feature points in (a) (by Ferns[2]) without taking into account the pose of the patch
- Rough estimation of the pose of the patch in (b) is limited to orientations

Solution (Panter)

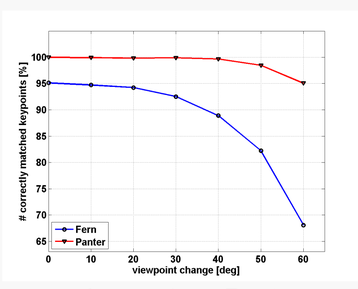
- No pre-classification but simultaneous estimation of keypoint identity and pose
- Estimation of the initial rectification not only from a small set of orientations but from a much larger set of real homography transformations



Patches extracted with Leopar[1]



Patches extracted with Panter



Leopar[1] vs. Panter

Results

- Fast (~8-17fps) and very accurate tracking by detection
- More robust to large perspective distortions and scale changes than Leopar[1]
- Only little texture and few feature points necessary to estimate the pose

More robust tracking by detection with better patch retrieval than Leopar [CVPR'08]

!!!! Pose estimation possible with only ONE feature point!!!

Simultaneous Estimation of Patch Identities and Pose

I. Extraction of Rough Rectification Information

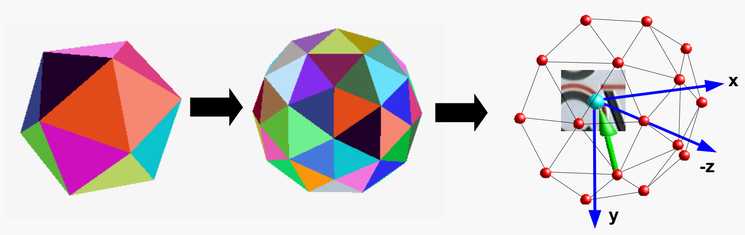
Given an image patch $\mathbf{p}_{i',j'}$ of Keypoint ***i'*** under Pose ***j'*** we want to find ***i'*** and ***j'*** (using linear classifiers for efficiency). Therefore, in the first step we assume that ***i'*** = ***i:***

$$\forall j' : a_{i,j}^T p_{i,j'} = \begin{cases} +1 : j = j' \\ -1 : otherwise \end{cases}$$

$\mathbf{a}_{i,j}$ can be learned in the training phase by a simple linear least squares method:

$$a_{i,j} = (PWW^T P^T)^{-1} PWW^T y$$

where ***P*** is the matrix made of patches column vectors samples $\boldsymbol{p}_{i,j}$, ***y*** is the row vector made of +1 and -1 values and ***W*** is a diagonal matrix containing the equation weights $\boldsymbol{w_+ = N-1}$ for positive and $\boldsymbol{w_- = 1}$ for negative examples under pose ***j*** (***N*** is the number of poses).
To ensure a equal distribution of the training samples of all important viewpoints we approximate a regular polyhedron with an initial icosaeder. The vertices of the approximated regular polyhedron serve as discretized viewpoints.



Training samples are only taken from a local neighborhood of the vertices.

At run-time we get for each patch $\mathbf{p}_{i',j'}$ and for each keypoint ***i*** in the database a list $\boldsymbol{\Gamma_i}$ of possible pose indices. We select for each keypoint ***i*** the best pose ***j*** that maximizes:

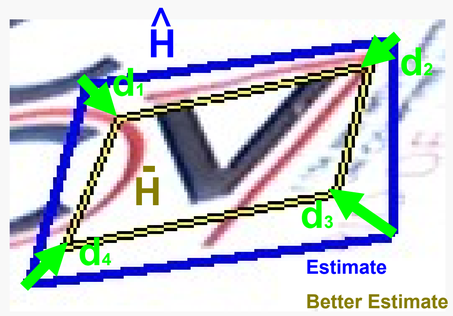
$$\forall i : \arg\max_{j \in \Gamma_i} p^T \bar{p}_{i,j}$$

where $\bar{p}_{i,j}$ is the mean of all positive examples of the patch of keypoint ***i*** under the pose ***j***.

II. Iterative Refinement with Linear Predictors[3]

For each keypoint ***i*** in the database and its corresponding best rough pose estimate ***j*** we try to refine the patch rectification:

$$\begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix} = B_i(p(H_j) - p_i^*)$$



where $\boldsymbol{H_j}$ is the initial homography estimate with respect to keypoint ***i***, $\boldsymbol{p(H_j)}$ the normalized intensity value vector of the patch under matrix $\boldsymbol{H_j}$ and $\boldsymbol{p_i}$* the normalized intensity vector of the reference patch. This equation has to be applied iteratively to converge to the right solution.

Matrix $\boldsymbol{B_i}$ can simply be learned by warping the patch of interest by small random pose changes ***δR*** and computing the normalized intensity changes ***δP***.

$$B_i = XD^T(DD^T)^{-1}$$

X: the training matrix with the **δR** vectors
D: the training matrix with the **δP** vectors

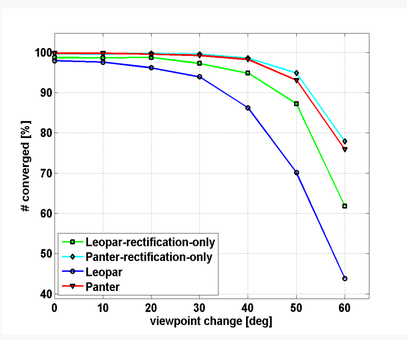
III. Establishing Patch Identity & Pose

Thanks to the accuracy of the Linear predictors, we can remove the outliers by simple cross-correlation between the warped patch and the reference patch:

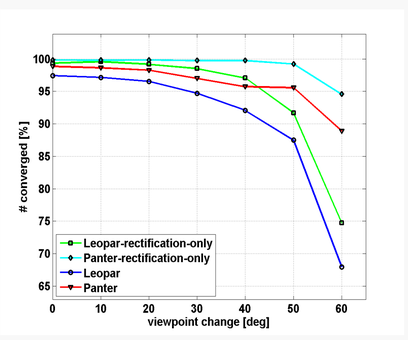
$$p(H_{final,i})^T \cdot p_i^* \geq \tau_{ncc}$$

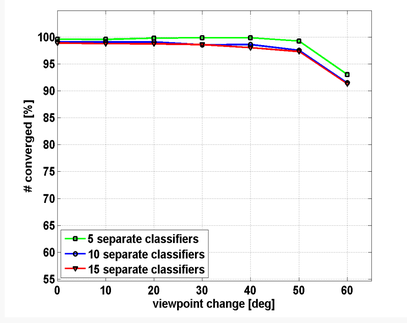
where $\boldsymbol{H_{final,i}}$ is the final transformation obtained with the linear predictor and ***i*** is the corresponding Keypoint identity. In practice we use a threshold $\boldsymbol{\tau_{ncc} = 0.9}$.

Experiments & Robust Real-Time Tracking by Detection

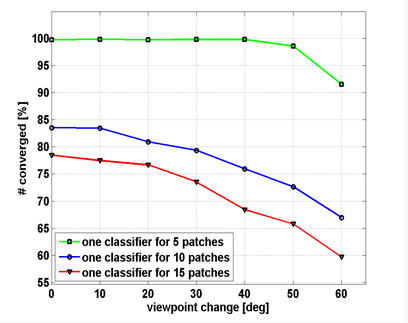


Panter outperforms Leopar[1] with respect to robustness against increasing viewpoint changes.





The performance of building one classifier for many patches is much lower than building one classifier for each patch.



The pose estimation with Panter is robust to high perspective distortions, to scale changes, to occlusion and even to some deformations.

References

[1] Hinterstoisser, S. et al.: Online Learning of Patch Perspective Rectification for Efficient Object Detection, CVPR 2008, Anchorage, Alaska, USA.
[2] Ozuysal, M. et al.: Fast Keypoint Recognition in Ten Lines of Code, CVPR 2007, Minneapolis, Minnesota, USA.
[3] Jurie, F. et al.: Hyperplane approximation for template matching, PAMI, 2002