Radon space and Adaboost for Pose Estimation

Patrick Etyngier[1] Nikos Paragios[2] Renaud Keriven[1] Yakup Genc[3] Jean-Yves Audibert[1]

[1]CERTIS Laboratory
Ecole des Ponts, Paris, France
etyngier@certis.enpc.fr

[2]MAS Laboratory
Ecole Centrale Paris, France
nikos.paragios@ecp.fr

[3]Siemens Corporate Research
Princeton NJ, USA
yakup.genc@siemens.com

Abstract

In this paper, we present a new approach to camera pose estimation from single shot images in known environment. Such a method comprises two stages, a learning step and an inference stage where given a new image we recover the exact camera position. Lines that are recovered in the radon space consist of our feature space. Such features are associated with [AdaBoost] learners that capture the wide image feature spectrum of a given 3D line. Such a framework is used through inference for pose estimation. Given a new image, we extract features which are consistent with the ones learnt, and then we associate such features with a number of lines in the 3D plane that are pruned through the use of geometric constraints. Once correspondence between lines has been established, pose estimation is done in a straightforward fashion. Encouraging experimental results based on a real case demonstrate the potentials of our method.

1. Introduction

Pose estimation has been extensively studied in the past years. Nevertheless, it is still an open problem particularly in the context of real time vision. Robot navigation, autonomous systems and self-localization are some of the domains in computational vision where pose estimation is important. In prior literature pose estimation methods are either feature-driven [9] or geometry-driven [1, 8, 7, 2]. In this paper, we aim to combine both approaches by considering geometric elements such as lines to be the most appropriate feature space. Indeed, lines are simple geometric structures that refer to a compact representation of the scene, while at the same time one can determine angles and orientations that relate their relative positions. Last but not least, appropriate feature spaces and methods exist for fast line extraction and manipulation (Hough[5, 10], Radon [10]).

Our method consists of a learning and an inference steps. During the learning stage, the scene is learnt from an image sequence and its corresponding 3D reconstruction. A geometry-based learning is achieved by recovering geometric relations between lines and consequently between their projections. In parallel to the feature-based learning, 3d lines are associated through AdaBoost learners with their 2D projection in the Radon space (local maxima). This information space is used within a matching process to recover camera's pose from a new image. Matching between plausible line candidates in a new image dictates multiple correspondences between the 2D new image lines and the 3D reconstructed lines. The most probable configuration in terms of appearance provides the camera position while geometric consistency constraints are satisfied. The overview of the approach is shown in [Fig. (1)].

The reminder of the paper is organized in the following fashion. In section 2 we state the problem and discuss feature detection through an image sequence as well as feature modeling. Pose estimation through inference is part of section 3, while experimental results based on a real case and discussion are presented in the last section.

2.1 Problem Formulation & Radon Spaces

Let us assume that the image place is perpendicular to the view axis. Using the perspective model, the image of any point in space is equal to the intersection of the image plane and the line joining the point to the center of the camera lens.

The main stream of research in 3D reconstruction and pose estimation has been devoted to point correspondences [9]. Line correspondences could be an efficient alternative to such an approach [7]. Such a feature space inherits the advantage of being more robust than point correspondences as well as more global. In recent years the Hough transform and the related Radon transform became very popular tools in image analysis and medical imaging. These two operators are able to transform a two dimensional image with lines into a domain of line parameters, where each line in

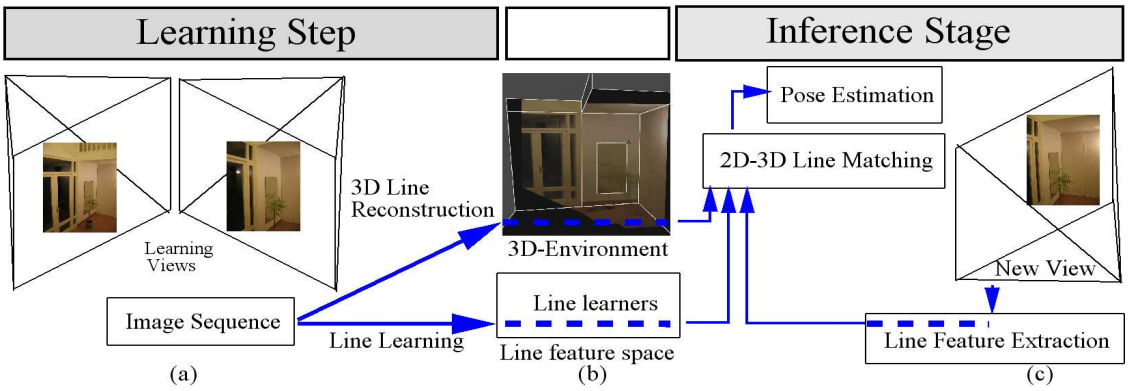

Figure 1. Overview of the proposed pose estimation approach where both learning and estimation steps are delineated.

the image will give a peak positioned at the corresponding line parameters.

Several definitions of the Radon transform exist. A very popular form expresses lines in the form $\rho = x * \cos(\theta) + y * \sin(\theta)$ where θ is the angle and ρ the smallest distance to the origin of the coordinate system. The Radon transform for a set of parameters (ρ, θ) is the line integral through the image $f(x, y)$, where the line is positioned corresponding to the value of $g(\rho, \theta)$:

$$g(\rho, \theta) = \iint_{\mathbb{R}^2} f(x, y)\delta(\rho - x\cos(\theta) - y\sin(\theta))dxdy$$

with $\delta()$ being the Dirac function. Local maxima in such a space correspond to lines in the original image and can be extracted in a straightforward fashion. This global transformation encodes the entire line structure in a compact fashion, and is capable to account for occlusions while local and global changes of the illumination as well as strong presence of noise can be dealt with. Since all projected lines in the image sequence have to be matched together, we proposed previously either to achieve such a task semi-automatically in case of image sequence, either to track lines in case of video sequence in the corresponding Radon spaces.

2.2 3D-2D Line Relation through Boosting

Once the scene and 3D lines have been reconstructed (central image in [Fig. 1]), one would like to establish a connection between such 3D lines and their corresponding projections. Since our approach is both features and geometric based, we aim at learning both kind of constraints.

First, geometrical constraints can be straight and naturally deduced from the 3D reconstructed scene implying 2d constraints on the projected lines. Since extraction of the relative geometry is not critical - once 3D reconstruction has been completed -, more attention is to be paid on feature extraction, learning and modeling.

Let us consider that our feature learning stage consists of $\mathcal{L} = \{l_1, l_2, ..., l_n\}$ 3D lines, and our training consists of c images. Without loss of generality we assume that such geometric elements were successfully detected within this c images. Let $\mathcal{P}_k = \{p_k^1, p_k^2, ..., p_k^c\}$ be the projections in the radon space of line l_k at these c images. Such projections correspond to the 2D local radon patches represented as d-dimensional vectors.

Traditional statistical inference techniques can be used to recover a distribution of such d-dimensional vectors, with d the number of pixel in the local patch. To this end, one can consider simple Gaussian assumptions and classical dimensionality reduction techniques like principal component analysis. Such a selection could fail to account for the highly non-linear structure of the Radon space and so of the corresponding features. Furthermore, since recovering a training shots from all possible virtual positions of the observer it is almost impossible, one should also account for sparse observations and learning from small training sets. Therefore, more advanced classification techniques that are able to cope with some of the above limitations are to be considered.

Our basic classifier consists of given two classes $\mathcal{C}_1$ and $\mathcal{C}_2$ find an appropriate transformation/function F that can measure the distance between a sample p and these classes $F(C_k, p)$. To this end, within the context of our application one can consider n bin classification problems F_k,

$$F_k(p) = \begin{cases} 1, & p \in \mathcal{C}_k \\ 0, & p \in \mathcal{C}_j,\ j \neq k \end{cases}$$

In other words, we are looking for a way to compute the

boundary of a binary partition between the features corresponding to line l_k versus the others. Stump classification can deal with this problem: it tests binary partitions along all the d dimensions and all possible thresholds. The model is given by:

$$R = \{\alpha_0 \mathbb{1}_{x_j<\tau} + \alpha_1 \mathbb{1}_{x_j \geq \tau} : j \in 1,\dots,d, \tau \in \mathbb{R}, \quad \alpha_0 \in [0;1], \alpha_1 \in [0;1]\} \tag{1}$$

The threshold τ^* and the dimension j^* that minimize the desired criteria $\mathcal{W}(j,\tau)$ are kept to form the partition parameters. The reader can refer to [4] to get further details about stumps and more particularly about the criteria $\mathcal{W}$ we used. Consequently, stump classification returns a function f_m that defines a partition of the space according to an hyperplane which is orthogonal to the canonical basis of $\mathcal{X}$:

$$f_m = f_{m,<}\mathbb{1}_{x \in \mathcal{X}^{<}_{j,\tau}} + f_{m,\geq}\mathbb{1}_{x \in \mathcal{X}^{\geq}_{j,\tau}} \tag{2}$$

Implementation of stumps has been done and tests with a synthetic data set showed they can be used as "weak" learners to be plugged in an AdaBoost [6] procedure to form an accurate classifier.

The general idea of boosting is to **1-** repeatedly use a "weak" learner [stumps returning a regression function f_m in our case] with some weights w_i^m on the training data - *m beeing the iteration index* - **2-** focus on misclassified data from one iteration to the next through the update of w_i^m:

$$w_i^m = \frac{w_i^{m-1} e^{-Y_i f_m(X_i)}}{\mathrm{K}} \quad \begin{matrix}\forall i \in \{1,\dots,N\} \\ \text{K: normalizing constant}\end{matrix} \tag{3}$$

where Y_i is the classication corresponding to the feature X_i, (X_i, Y_i) beeing an element of the learning and N its size.

Then, at each step a weight c_m associated with the current learner is determined according to the corresponding classification performance. The final classification is given by the thresholded regression function $\mathbb{1}_{G_M(x)>T}$, $G_M(x)$ beeing the weighted combination of the "weak" learners:

$$G_M(x) = \sum_{m=1}^{M} c_m f_m \tag{4}$$

$G_M(x)$ is by definition piece-wise constant, the threshold T is thus chosen among the finite set of possible values so that the error classification is decreased.

The feature learning stage outputs n classifiers $\mathcal{S}^n = \{\mathbb{1}_{G^1_M(x)>T_1}, \dots, \mathbb{1}_{G^k_M(x)>T_k}, \dots \mathbb{1}_{G^n_M(x)>T_n}\}$ *-one for each line-* that are going to be used for line inference and pose estimation.

3. Line Inference & Pose Estimation

Line inference consists of recovering the most probable 2D patches-to-3D lines configuration using the set of classifiers $\mathcal{S}^n$ In this section, we first explore the straighforward solution and then we propose an objective function that couples the outcome of the Adaboost learners with geometric constraints inherited from the learning stage. In order to validate the performance of the AdaBoost classifier, we have created a realistic synthetic environment. The feature vector for one preselected line has been learnt, and the corresponding classifier was tested with new images: learning error converges to zero while the error of the classification in the test remains low and stable as the number of iteration increase. This remark is consistent with the expected behavior of the classifier; boosting does not overfit. As for testing error, samples from Class $\mathcal{C}_2$ are almost never misclassified while classification error of Class $\mathcal{C}_1$ is not low enough to give sufficiently confidence in line 2D-3D matching for pose estimation.

Such a limitation can be dealt with the use of geometrical constraints encoded in the learning state during the 3D reconstruction step. This assumption could allow us to relax the AdaBoost, since classification errors become less significant once geometry is introduced. A modified classification model is now constructed based on the previous observations. Let j be a new image. Any sample p such that $\left(G^k_M(p) > T_k\right)$ (Class $\mathcal{C}_1$) is a potential match. Moreover, classification confidence depends on the distance of the data to be classified from the boundary and so on the value of $\mathrm{sd}^k(x) = G^k_M(x) - T_k$: the greater is $|\mathrm{sd}^k(x)|$ the more confident is the classification. Thus, the easiest classification choice is:

$$\arg \max_{i \in \{1,\dots,n\}} \left(G^k_M(p^j_i) - T_k\right) \quad \text{s.t.} \quad G^k_M(p^j_i) > T_k \tag{5}$$

The correspondance expressed in eqn. (5) is not sufficient since the most important value does not necessarily correspond to the real match. Let us assume for a line k, we are interested in the B best potential matches $\{p_{n_1}[k], \dots, p_{n_B}[k]\}$. Such candidates are determine through the eqn. (5). If less than B lines verify the constraint $G^k_M(p_i[k])) > T_k \, \forall i$, then it is "relaxed" as earlier explained. In others words, lines misclassified are authorized to be taken into consideration by removing the constraint in eqn. (5). A weighting function $h(.)$ is also used to influence the importance of a potential match based on the quantity $\mathrm{sd}^k(.)$.

Now we want to express a geometrical constraint GC between the projections of C lines $\{l_{s_1}, \dots, l_{s_c}, \dots, l_{s_C}\}$ ($C < B$). For each lines s_c we keep the B best potential matches $\{p_{n_1}[s_c], \dots, p_{n_b}[s_c], \dots, p_{n_B}[s_c]\}$. Finally, the energy to be minimized is given by:

$$\min_{\substack{(i_1 \dots i_C) \in \\ (\mathcal{A}_1,\dots,\mathcal{A}_C)}} \sum_{c=1}^{C} h(\mathrm{sd}^{i_c}(p_{i_c}[s_c])) \quad \text{s.t. } \mathrm{GC}(p_{i_1}[s_1], \dots, p_{i_C}[s_C]) \tag{6}$$

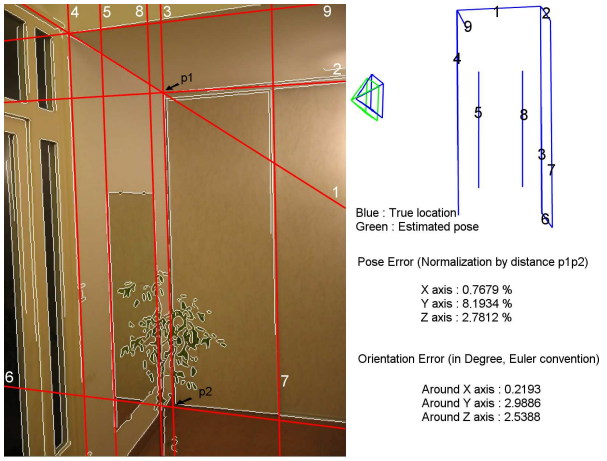

Figure 2. Final calibration: the image to be calibrated is overlayed by the edge map (in white) and the 3D line reprojection (in red)

where $\mathcal{A}_c$ is the indice set of potential matches with line l_{s_c}.One can recover the lowest potential of such a cost function using classical optimization methods. At the sight of the small number of lines detected, we consider an exaustive search approach. Numerous formulations can be considered for the GC term. Corners are prominent characteristics of 3D scenes. Therefore, 3D lines going through the same point (that can also define an orthogonal basis) is a straighforward geometry-driven constraint. One can use this assumption to define constraints in their projection space; that is:

$$\mathrm{GC}(l_1, l_2, l_3) = |(l_1 \times l_2)^T \, l_3| \qquad \begin{matrix} \times\text{: cross product} \\ T\text{: Transpose sign} \end{matrix} \qquad (7)$$

This term takes into account the scene context. Offices, buildings, etc. are scenes where the use of such a constraint is mostly justified (corners, vanishing points etc ...). For example in figure 2, the learning step of lines 1,2 and 3 gives a set $\left\{ \mathbb{1}_{G^1_M(x)>T_1}, \mathbb{1}_{G^2_M(x)>T_2}, \mathbb{1}_{G^3_M(x)>T_3} \right\}$. If only feature constraint is used through eqn. 5, only line 2 is well matched. However, by using relaxation and the geometrical constraint associated to these lines, the algorithm retrieves the good matching. In more complex scenes, more advanced terms can be considered to improve the robustness of the method. Once the line correspondence problem has been solved, we used the efficient method described in [3] to determine pose parameters of the camera.

4. Discussion

Several experiments were conducted to determine the performance of the method. To this end, first lines from an image sequence are matched and reconstructed -through standard method- Such a model refers to n classifiers with their features space being patches of the radon transformation of the original image sequence. Then, a new image of the same scene was considered and self-localization of the observer based on 2d-3d line matching [Fig. (2)] was performed.

In this paper, we have proposed a new technique to pose estimation from still images in known environments. Our method comprises a learning step where a direct association between 3D lines and radon patches is obtained. Boosting is used to model that statistical characteristics of these patches. Such a classification process provides multiple possible matches for a given line and therefore a fast prunning technique that encodes geometric consistency in the process is proposed. Such additional constraints overcome the limitation of classification errors and increase the performance of the method. Once the learning is done, inference stage of boosting is very fast, and we used moreover a linear fast 2d-3d calibration based on lines [3]. Better classification and more appropriate statistical models of lines in radon space is the most promissing direction. The use of radon patches encodes to some extend clutter. Therefore separating lines from irelevant information could improve the performance of the method.

References

[1] O. Ait-Aider, P. Hoppenot, and E. Colle. Adaptation of lowe's camera pose recovery algorithm to mobile robot self-localisation. *Robotica*, 20(4):385–393, 2002.

[2] P. Allen, A. Troccoli, B. Smith, S. Murray, I. Stamos, and M. Leordeanu. New methods for digital modeling of historic sites. *IEEE Comput. Graph. Appl.*, 23(6):32–41, 2003.

[3] A. Ansar and K. Daniilidis. Linear pose estimation from points or lines. In *ECCV 2002*, pages 282–296, 2002.

[4] J.-Y. Audibert. Aggregated estimators and empirical complexity for least square regression. In *Ann. Inst H. Poincaré*, volume 40, pages 685–736, Nov–Dec 2004.

[5] R. Duda and P. Hart. Use of the hough transformation to detect lines and curves in pictures. *Com. ACM*, 15(1), 1972.

[6] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *ICML*, pages 148–156, 1996.

[7] S. C. Lee, S. K. Jung, and R. Nevatia. Automatic pose estimation of complex 3d building models. In *WACV*, 2002.

[8] T. Phong, , R. Horaud, A. Yassine, and P. Tao. Object pose from 2d to 3d point and line correspondences. *IJVC*, 15(3):225–243, July 1995.

[9] E. Royer, M. Dhome, M. Lhuillier, and T. Chateau. Localization in urban environments: Monocular vision compared to a differential gps sensor. In *CVPR (2)*, pages 114–121, 2005.

[10] M. van Ginkel, C. L. Hendriks, and L. van Vliet. A short introduction to the radon and hough transforms and how they relate to each other. Technical Report QI-2004-01, Quantitative Imaging Group, Delft University of Technology, 2004.