Leveraging Feature Uncertainty in the PnP Problem

Luis Ferraz[1]
luis.ferraz@upf.edu
Xavier Binefa[1]
xavier.binefa@upf.edu
Francesc Moreno-Noguer[2]
fmoreno@iri.upc.edu

[1] Universitat Pompeu Fabra (DTIC)
08018, Barcelona, Spain

[2] Institut de Robòtica i Informàtica
Industrial (CSIC-UPC)
08028, Barcelona, Spain

Abstract

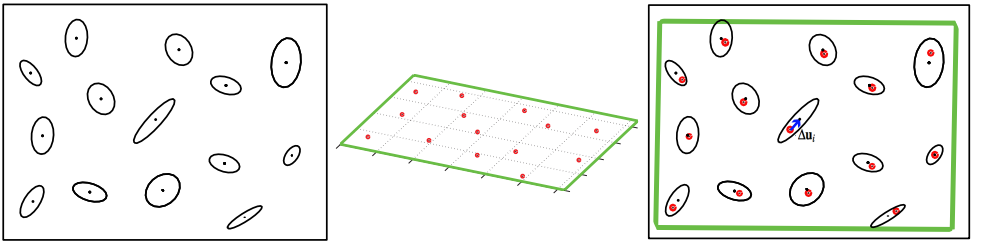
We propose a real-time and accurate solution to the Perspective-n-Point (P*n*P) problem –estimating the pose of a calibrated camera from *n* 3D-to-2D point correspondences– that exploits the fact that in practice the 2D position of not all 2D features is estimated with the same accuracy. Assuming a model of such feature uncertainties is known in advance, we reformulate the P*n*P problem as a maximum likelihood minimization approximated by an unconstrained Sampson error function, which naturally penalizes the most noisy correspondences. The advantages of this approach are thoroughly demonstrated in synthetic experiments where feature uncertainties are exactly known.

Pre-estimating the features uncertainties in real experiments is, though, not easy. In this paper we model feature uncertainty as 2D Gaussian distributions representing the sensitivity of the 2D feature detectors to different camera viewpoints. When using these noise models with our PnP formulation we still obtain promising pose estimation results that outperform the most recent approaches.

1 Introduction

The goal of the Perspective-*n*-Point (P*n*P) problem is to estimate the position and orientation of a calibrated camera from a set of *n* correspondences between 3D points and their 2D projections. It is a problem that lies at the core of a large number of applications in computer vision, augmented reality, robotics and photogrammetry.

Yet, while the P*n*P problem has been studied for more than a century, recent works have shown impressive results in terms of accuracy and efficiency. For instance, the Efficient P*n*P (EP*n*P) [23] was the first closed-form solution to the problem with $O(n)$ complexity and almost no loss of accuracy with respect to the most accurate iterative methods existing at the moment [21]. Subsequent works [11, 19, 33, 34] have improved, also with $O(n)$ complexity, the accuracy of the EP*n*P specially for the minimal cases with $n = \{3,4,5\}$ correspondences. In addition, the flexibility of the linear formulation of the EP*n*P has allowed reformulations of the problem to also estimate the intrinsic parameters of an uncalibrated camera [26], or to integrate an algebraic outlier rejection criterion which does not require executing multiple RANSAC iterations [5].



(a) (b) (c)

Figure 1: PnP problem with noisy correspondences. We assume a set of 2D feature points are given, with particular noise models for each of them, as shown in (a). We also assume the correspondences (red dots) with respect to a 3D model are known, as shown in (b). Our approach estimates a solution of the PnP problem that minimizes the Mahalanobis distances $\Delta\mathbf{u}_i$ shown in (c). In (c) green rectangle and red dots are the true projection of the 3D model.

In any event, although all previous P*n*P solutions assume that correspondences may be corrupted by noise and show robustness against large amounts of it, none of these works considers that the particular structure of the uncertainty associated to each correspondence could indeed to be used to further improve the accuracy of the estimated pose (see Fig. 1). Specifically, existing solutions assume all 2D correspondences to be affected by the same model of noise, a zero mean Gaussian distribution, and consider all correspondences to equally contribute to the estimated pose, independently of the precision of their actual location.

In contrast, in this paper we propose a solution to the P*n*P problem which, to the best of our knowledge, is the first one that inherently incorporates into its formulation feature uncertainties. We do this by iteratively minimizing an uncontrained Sampson error function [10], which approximates the Maximum Likelihood solution. Furthermore, we also propose a strategy to compute a specific uncertainty model per correspondence in real experiments, by modeling the sensitivities of 2D detectors to different viewpoints. As we will show in both synthetic and real experiments, our approach outperforms the most recent techniques in terms of accuracy while keeping a running time still linear with respect to the number of correspondences.

2 Related work

Traditionally, the P*n*P problem has been applied to small subsets of correspondences yielding closed form solutions to the P3P [4, 7, 9, 16], P4P [6], and P5P [32] problems. Yet, these solutions to the minimal case are prone to be sensitive to noise, being therefore typically used within RANSAC schemes. Noise robustness can be achieved by considering larger sizes of the correspondence set. For uncalibrated cameras, the most straight-forward algorithm for doing so, is the Direct Linear Transformation (DLT) [10].

When the internal parameters of the camera are known, there exist iterative P*n*P approaches which optimize an objective function involving an arbitrary number of correspondences. Standard objective functions are based on geometric (*e.g.* 2D reprojection) [25] or algebraic errors [21]. Yet, iterative methods suffer from a high computational cost and tend to be sensitive to local minima [8, 21]. Paradoxically, early non-iterative solutions were neither computationally tractable, as they considered all n points as unknowns of the problem [1, 27].

This has recently been overcome by a series of $O(n)$ formulations that can afford arbitrarily large point sets. The first of these techniques was the EPnP [18, 23], that reduced the PnP to retrieving the position of four control points spanning any number n of 3D points. This reformulation of the problem, jointly with the use of linearization strategies, permitted dealing with hundreds of correspondences in real time. The EPnP has been revisited in [5], where the problem is reformulated in terms of an Efficient Procrustes PnP (EPPnP), yielding to even additional speed-ups. Subsequent works have improved the accuracy of the EPnP, still in $O(n)$, by replacing the linearization with polynomial solvers. The most remarkable works along these lines are the Robust PnP (RPnP) [19], the Direct-Least-Squares (DLS) [11], the Accurate and Scalable PnP (ASPnP) [32] and the Optimal PnP (OPnP) [33].

Yet, as we have pointed out above, the noise problem has not been directly handled by previous PnP solutions, which simply attenuate its effect by exploiting data redundancy. In contrast, other problems in geometric computer vision, such as Simultaneous Pose and Correspondence [24, 28, 31], Fundamental matrix computation [2, 14] , ellipse fitting [2, 13, 15, 17], do take into account specific models of uncertainty per observed point. In most these approaches, the uncertainty is modeled by a covariance matrix, and Maximum Likelihood strategies are proposed to minimize the Mahalanobis distances between the noisy and the true locations of the point observations. As discussed in [3], estimating the global minima for this kind of problems is impractical. A feasible alternative is to minimize approximated Sampson error functions, for instance by means of iterative approaches such as the Fundamental Numerical Scheme (FNS) [2], the Heterocedastic Errors-in-Variables (HEIV) [17] or projective Gauss Newton [14]. These minimization approaches can be considered as a solution refinement and they need to be fed with an initial solution. Other methods as the Renormalization [12] or the recent Hyper-Renormalization [15] do not need an initial solution, and find a solution by solving a set of *estimating equations* which need not to be derivatives of some cost function.

All these previous approaches, though, are focused on theoretical derivations which are only evaluated over synthetic data where the uncertainty models per point are assumed to be known in advance. The problem of estimating these input models in real data is completely obviated. In this paper we will propose a strategy for this in the case of estimating the pose of planar objects.

3 Covariant EPPnP

We next describe our PnP approach that exploits prior information about the uncertainty in the location of image features, as well as the approach to estimate these uncertainty models on real images. More specifically, we first state the PnP problem and review the EPPnP [5] linear formulation. Then, we reformulate the EPPnP in order to integrate feature uncertainties and estimate the camera pose based on an approximated Maximum Likelihood procedure. And finally, we describe a methodology to model viewpoint-independent 2D feature uncertainties using anisotropic Gaussian distributions.

3.1 Problem statement and EPPnP Linear Formulation

Let us assume we are given a set of 3D-to-2D correspondences between n 3D reference points $\mathbf{p}_i^w = [x_i^w, y_i^w, z_i^w]^\top$ expressed in a world coordinate system w and their 2D projections $\mathbf{u}_i = [u_i, v_i]^\top$. Let $\mathbf{A}$ be the camera internal calibration matrix, also assumed to be known in

advance. Using these assumptions, the goal of the PnP is to estimate, using a single image, the rotation matrix $\mathbf{R}$ and translation $\mathbf{t}$ that align the camera and world coordinate frames. For each 2D feature point i, we have the following perspective constraint:

$$d_i \begin{bmatrix} \mathbf{u}_i \\ 1 \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{R} | \mathbf{t} \end{bmatrix} \begin{bmatrix} \mathbf{p}_i^w \\ 1 \end{bmatrix}, \tag{1}$$

where d_i is the depth of the feature point. Following the EPnP [23] derivation, $\mathbf{p}_i^w$ can be rewritten in terms of the barycentric coordinates of four control points $\mathbf{c}_j^w$, $j = 1,\dots,4$, chosen so as to define an orthonormal basis centered at the origin of the world coordinate system. Every reference point, can therefore be expressed as $\mathbf{p}_i^w = \sum_{j=1}^4 \alpha_{ij} \mathbf{c}_j^w$.

Note that the barycentric coordinates α_{ij} are independent on the coordinate system, and specifically they remain the same when writing the reference points in the camera coordinate system c. That is, $\mathbf{p}_i^c = \sum_{j=1}^4 \alpha_{ij} \mathbf{c}_j^c$.

From the EPPnP [5], after some operations and matrix manipulations, Eq. 1 can be rewritten as the following Kronecker product:

$$\begin{bmatrix} \alpha_{i1} & \alpha_{i2} & \alpha_{i3} & \alpha_{i4} \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & -u_i^c \\ 0 & 1 & -v_i^c \end{bmatrix} \mathbf{x} = \mathbf{0} \tag{2}$$

where $[u_i^c, v_i^c, 1]^\top = \mathbf{A}^{-1} [u_i, v_i, 1]^\top$ are the normalized 2D coordinates and $\mathbf{x} = [\mathbf{c}_1^{c\top}, \dots, \mathbf{c}_4^{c\top}]^\top$ is the unique unknown, a 12-dimensional vector containing the control point coordinates in the camera reference system.

Finally, the concatenation of Eq. 2 for all n correspondences can be expressed as a linear system $\mathbf{M}\mathbf{x} = \mathbf{0}$ where $\mathbf{M}$ is a $2n \times 12$ known matrix.

The ultimate goal is to estimate the $\mathbf{R}$ and $\mathbf{t}$, which provide the absolute camera pose in the world coordinate system. However, $\mathbf{x}$ is an estimation of the subspace where the control points $\mathbf{c}_j^c$ in camera referencial lies, *i.e.* any scaled version $\gamma\mathbf{x}$ would also be a solution of Eq. 2. The EPPnP [5] proposes to estimate $\mathbf{R}$, $\mathbf{t}$ and γ in closed-form using a generalization of the Orthogonal Procrustes problem [29]:

$$\underset{\gamma,\mathbf{R},\mathbf{t}}{\arg\min} \sum_{j=1}^4 \| \mathbf{R}\mathbf{c}_j^w + \mathbf{t} - \gamma \mathbf{c}_j^c \|^2 \qquad \text{subject to } \mathbf{R}^\top \mathbf{R} = \mathbf{I}_3 \tag{3}$$

Additionally, [5] also proposes an iterative refinement of $\mathbf{x}$. Considering the control points positions estimated from Eq. 3, $\hat{\mathbf{c}}_j^c = \mathbf{R}\mathbf{c}_j^w + \mathbf{t}$, the vector $\hat{\mathbf{x}} = [(\hat{\mathbf{c}}_1^c)^\top, \dots, (\hat{\mathbf{c}}_4^c)^\top]^\top$ is obtained and projected on the extended null-space of $\mathbf{M}$, built using its 4 eigenvectors with smaller singular values. The projected $\hat{\mathbf{x}}$ is in turn fed again into Eq. 3 to recompute γ, $\mathbf{R}$ and $\mathbf{t}$. This process is followed until convergence.

Just as with the EPnP [23], the planar case requires a slight modification of the method. Since in this case only three control points are necessary to span the reference points onto the plane, the dimensionality of our vector of unknowns $\mathbf{x}$ drops to 9, and $\mathbf{M}$ becomes a $2n \times 9$ matrix of correspondences. Besides these changes the rest of the algorithm remains completely unchanged.

3.2 Integration of Feature Uncertainties in the Linear Formulation

Let us assume that $\mathbf{u}_i = [u_i, v_i]^\top$ in Eq. 1 represents an observed 2D feature location obtained using a feature detector. This observed value can be regarded as a perturbation from its true

2D projection $\bar{\mathbf{u}}_i$ by a random variable $\Delta\mathbf{u}_i$. We write this as,

$$\mathbf{u}_i = \bar{\mathbf{u}}_i + \Delta\mathbf{u}_i \tag{4}$$

We assume that $\Delta\mathbf{u}_i$ is small, independent and unbiased allowing to model the uncertainty in statistical terms, with expectation $E[\Delta\mathbf{u}_i] = \mathbf{0}$ and covariance $E[\Delta\mathbf{u}_i\Delta\mathbf{u}_i^\top] = \sigma^2\mathbf{C}_{\mathbf{u}_i}$, where $\mathbf{C}_{\mathbf{u}_i}$ is the known 2×2 uncertainty covariance matrix and σ is an unknown global constant specifying the global uncertainty in the image. Splitting the uncertainty term into two components is motivated because the optimal solution can be obtained ignoring σ [13], making the known uncertainties to be independent of the object size in the image.

From these assumptions, the likelihood of each observed 2D feature location $\mathbf{u}_i$ from its true 2D projection $\bar{\mathbf{u}}_i$ can be expressed as,

$$P(\mathbf{u}_i) = k\cdot\exp(-\frac{1}{2}(\mathbf{u}_i-\bar{\mathbf{u}}_i)^\top\mathbf{C}_{\mathbf{u}_i}^{-1}(\mathbf{u}_i-\bar{\mathbf{u}}_i)) = k\cdot\exp(-\frac{1}{2}\Delta\mathbf{u}_i^\top\mathbf{C}_{\mathbf{u}_i}^{-1}\Delta\mathbf{u}_i) \tag{5}$$

where k is a normalization constant.

Thus, the Maximum Likelihood solution for the PnP problem is equivalent to minimizing the Mahalanobis distance in Eq. 5 for all n correspondences,

$$\underset{\Delta\mathbf{u}_i,\mathbf{x}}{\arg\min}\sum_{i=1}^{n}\|\Delta\mathbf{u}_i\|^2_{\mathbf{C}_{\mathbf{u}_i}^{-1}} \qquad \text{subject to} \quad \mathbf{M}_{\bar{\mathbf{u}}_i}\mathbf{x} = \mathbf{0} \tag{6}$$

where $\mathbf{M}_{\bar{\mathbf{u}}_i}\mathbf{x} = \mathbf{0}$ enforce the 3D-to-2D projective constraints in terms of the noise-free correspondences. Assuming the uncertainty $\Delta\mathbf{u}_i = [\Delta u_i \;\; \Delta v_i]^\top$ to be small, a first order perturbation analysis allows to approximate the projective constraint as,

$$\mathbf{M}_{\bar{\mathbf{u}}_i}\mathbf{x} = \mathbf{M}_{\mathbf{u}_i}\mathbf{x} - \Delta u_i\nabla_u\mathbf{M}_{\mathbf{u}_i}\mathbf{x} - \Delta v_i\nabla_v\mathbf{M}_{\mathbf{u}_i}\mathbf{x} = \mathbf{0} \tag{7}$$

where $\nabla_u\mathbf{M}_{\mathbf{u}_i}$ and $\nabla_v\mathbf{M}_{\mathbf{u}_i}$ are the partial derivatives of $\mathbf{M}_{\mathbf{u}_i}$ in Eq. 2 with respect to u and v,

$$\begin{aligned}\nabla_u\mathbf{M}_{\mathbf{u}_i} &= \begin{bmatrix}\alpha_{i1} & \alpha_{i2} & \alpha_{i3} & \alpha_{i4}\end{bmatrix}\otimes\begin{bmatrix}0 & 0 & -1\\ 0 & 0 & 0\end{bmatrix}\\ \nabla_v\mathbf{M}_{\mathbf{u}_i} &= \begin{bmatrix}\alpha_{i1} & \alpha_{i2} & \alpha_{i3} & \alpha_{i4}\end{bmatrix}\otimes\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & -1\end{bmatrix}\end{aligned} \tag{8}$$

Using Lagrange multipliers we can further eliminate the constraints in Eq. 6, and write the problem as an unconstrained minimization of the Sampson error function $J(\mathbf{x})$, namely

$$\underset{\mathbf{x}}{\arg\min}\; J(\mathbf{x}) = \underset{\mathbf{x}}{\arg\min}\sum_{i=1}^{n}\frac{\mathbf{x}^\top\mathbf{M}_{\mathbf{u}_i}^\top\mathbf{M}_{\mathbf{u}_i}\mathbf{x}}{\mathbf{x}^\top\mathbf{C}_{\mathbf{M}_i}\mathbf{x}}\,. \tag{9}$$

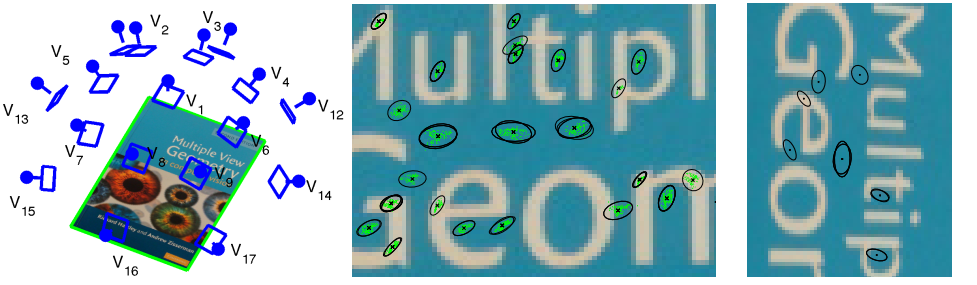
$\mathbf{C}_{\mathbf{M}_i}$ is the following 12×12 covariance matrix

$$\mathbf{C}_{\mathbf{M}_i} = (\nabla_u\mathbf{M}_{\mathbf{u}_i} + \nabla_v\mathbf{M}_{\mathbf{u}_i})^\top\mathbf{C}_{\mathbf{u}_i}(\nabla_u\mathbf{M}_{\mathbf{u}_i} + \nabla_v\mathbf{M}_{\mathbf{u}_i})\,, \tag{10}$$

which can be interpreted as the uncertainty $\mathbf{C}_{\mathbf{u}_i}$ propagated to the $\mathbf{M}$-space [13].

In order to minimize Eq. 9 we take the derivative of $J(\mathbf{x})$ with respect to $\mathbf{x}$:

$$\frac{\partial J}{\partial\mathbf{x}} = 2\sum_{i=1}^{n}\frac{\mathbf{M}_{\mathbf{u}_i}^\top\mathbf{M}_{\mathbf{u}_i}}{\mathbf{x}^\top\mathbf{C}_{\mathbf{M}_i}\mathbf{x}}\mathbf{x} - 2\sum_{i=1}^{n}\frac{\mathbf{x}^\top\mathbf{M}_{\mathbf{u}_i}^\top\mathbf{M}_{\mathbf{u}_i}\mathbf{x}\mathbf{C}_{\mathbf{M}_i}}{(\mathbf{x}^\top\mathbf{C}_{\mathbf{M}_i}\mathbf{x})^2}\mathbf{x} = \mathbf{N}\mathbf{x} - \mathbf{L}\mathbf{x} \tag{11}$$



(a) (b) (c)

Figure 2: Feature uncertainties on real images. (a) Example of a grid of reference views where uncertainties must be estimated. (b) Example of feature point clouds (in green) and their Gaussian models (black ellipses) for V_1. (c) Results of feature matching and uncertainties alignment against a test image.

where $\mathbf{N}$ and $\mathbf{L}$ are 12×12 matrices which also depend on $\mathbf{x}$.

Setting the previous equation to zero we obtain $\mathbf{N}\mathbf{x} - \mathbf{L}\mathbf{x} = \mathbf{0}$. As we mentioned in the related work section, there are several techniques to compute $\mathbf{x}$ from this equation, e.g [2, 14, 17]. In this paper, we chose the Fundamental Numerical Scheme (FNS) approach [2], which solves iteratively the following eigenvalue problem,

$$(\mathbf{N} - \mathbf{L})\mathbf{x} = \lambda \mathbf{x} \tag{12}$$

where the eigenvector with smaller eigenvalue is used to update the solution $\mathbf{x}$. Note that at each iteration, the matrices $\mathbf{N}$ and $\mathbf{L}$ need to be updated with the new $\mathbf{x}$ estimate up to the convergence. For the first iteration, $\mathbf{x}$ is initialized solving the equation system $\mathbf{M}\mathbf{x} = \mathbf{0}$ as is done in the EPPnP method.

Finally, once $\mathbf{x}$ is estimated, the PnP problem is solved using the generalized Orthogonal Procrustes problem of Eq. 3 and its refinement.

3.3 Dealing with Feature Uncertainties on Real Images

Estimating 2D feature uncertainties ($\mathbf{C}_{\mathbf{u}_i}$ in previous section) in real images is still an open problem. Most of previous approaches dealing with geometric estimation under noise, just address the problem in synthetic situations where 2D uncertainties are perfectly modeled using Gaussian distributions. In this paper, we propose an approach to model stochastically the behavior of a feature detection algorithm under real camera pose changes.

Our approach starts by detecting features on a given reference view $\mathbf{V}_r$ of the object of interest. Then, we synthesize m novel views $\{\mathbf{I}_1, \ldots, \mathbf{I}_m\}$ of the object, which sample poses around $\mathbf{V}_r$. Each view $\mathbf{I}_j$ is generated by projecting the 3D object model onto the image plane assuming a known projection matrix $\mathbf{P}_{\mathbf{I}_j} = \mathbf{A}[\mathbf{R}_{\mathbf{I}_j} | \mathbf{t}_{\mathbf{I}_j}]$ (without changing the distance between the 3D object center and the camera to avoid changes in the global scale of the uncertainty σ). We then extract 2D features for each $\mathbf{I}_j$, and reproject them back to $\mathbf{V}_r$, creating feature point clouds (see Fig. 2b).

Note that to compute these point clouds onto the reference view we have not performed any feature matching process,*i.e.*, there might be points within one cluster that do not belong to the same feature. In order to resolve this issue and perform a correct clustering of features we make use of the repeatability measure proposed in [22], which measures to what extend

do the area associated to each feature overlap. This area could correspond, for instance, to the scale ellipse defined by the SIFT detector [20]. Specifically, if we denote by μ_a and μ_b the two feature points, and $\mathbf{S}_{\mu_a}$ and $\mathbf{S}_{\mu_b}$ their corresponding image areas, we compute the following intersection over union measure:

$$\text{IoU} = \frac{\mathbf{S}_{\mu_a} \cap \mathbf{S}_{(\mathbf{H}^\top \mu_b \mathbf{H})}}{\mathbf{S}_{\mu_a} \cup \mathbf{S}_{(\mathbf{H}^\top \mu_b \mathbf{H})}} \tag{13}$$

where $\mathbf{H}$ is the known planar homography relating both regions, and $\cap$ and $\cup$ represent the intersection and union of the regions. Then, the two feature points are deemed to correspond if $1 - \text{IoU} < 0.4$, following the same criterion as in [22].

Once features are grouped we model each cluster i using a Gaussian distribution, with associated covariance matrix $\mathbf{C}_{\mathbf{u}_i}$. Note that this covariance tends to be anisotropic, which means that it is not rotationally invariant with respect to the roll angle. To achieve this invariance we use the angles of the main gradients of the feature regions, similarly as is done by the SIFT detector [20]. Fig. 2b and 2c show how each $\mathbf{C}_{\mathbf{u}_i}$ is rotated respect to the main feature gradients.

In practice, we found that $\mathbf{C}_{\mathbf{u}_i}$ describes with accuracy the uncertainties when the camera pose of $\mathbf{I}_j$ is near to the camera pose of the reference $\mathbf{V}_r$. This accuracy drops when camera pose moves away. This is motivated because each $\mathbf{C}_{\mathbf{u}_i}$ is computed for $\mathbf{V}_r$, remind that uncertainties are not on the 3D model. In order to handle this, we defined a set of l reference images $\{\mathbf{V}_1, ..., \mathbf{V}_l\}$ under different camera poses and each one with its own uncertainty models. We experimentally found that taking a grid of reference images all around the 3D object every 20° in yaw and pitch angles (see Fig. 2a), we obtained precise uncertainty models. Before running the Covariant EPPnP algorithm we have proposed, we had to choose an initial reference image to start with. For this, we used the EPPnP.

In summary, the algorithm for real images can be split into the following three main steps:

1. Estimate an initial camera pose without considering feature uncertainties using EPPnP. Let $[\mathbf{R}|\mathbf{t}]_{EPPnP}$ be this initial pose.
2. Pick the nearest reference view $\mathbf{V}_k$ taking into account that roll angle is not used to compute the grid of reference images. Find $\mathbf{V}_k$ such that

$$max_k \left(\frac{\mathbf{c}_k^\top}{\|\mathbf{c}_k\|} \cdot \frac{\mathbf{c}_{EPPnP}}{\|\mathbf{c}_{EPPnP}\|} \right) \tag{14}$$

   where $\mathbf{c}_k/\|\mathbf{c}_k\|$ and $\mathbf{c}_{EPPnP}/\|\mathbf{c}_{EPPnP}\|$ are the normalized camera centers in world coordinates, being $\mathbf{c}_k = -\mathbf{R}_k^\top \mathbf{t}_k$ and $\mathbf{c}_{EPPnP} = -\mathbf{R}_{EPPnP}^\top \mathbf{t}_{EPPnP}$.
3. Solve Eq. 12 using the covariances $\mathbf{C}_{\mathbf{u}_i}$ of the reference image $\mathbf{V}_k$, and $[\mathbf{R}|\mathbf{t}]_{EPPnP}$ for initializing the iterative process. The final pose $[\mathbf{R}|\mathbf{t}]_{CEPPnP}$ is obtained from Eq. 3 and its refinement.

4 Experimental results

We compare the accuracy and scalability of our method against state-of-the-art on synthetic and real data. Our method is implemented in MATLAB and the source code will be made publicly available in the authors webpage.

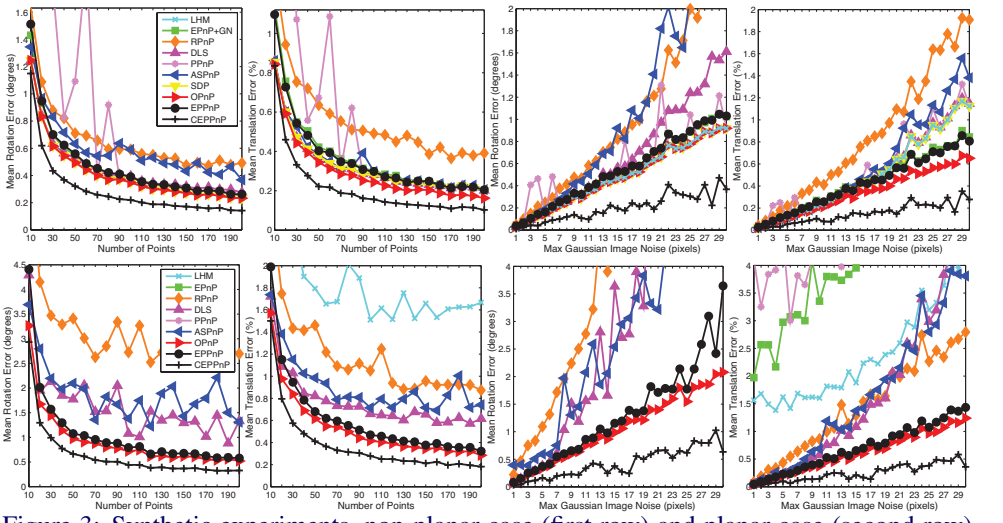


Figure 3: Synthetic experiments, non-planar case (first row) and planar case (second row), varying the number of correspondences and the uncertainty level.

4.1 Synthetic experiments

In these experiments we compared the accuracy and running time of our proposed method assuming each uncertainty has a known Gaussian distribution. In experiments we refer to our method as Covarinat Efficient Procrustes PnP (CEPPnP).

We have compared our formulations against the most recent PnP approaches: the robust version of DLS [11], ASPnP [34], OPnP [33], RPnP [19], PPnP [8], EPnP + GN [18], SDP [30], EPPnP [6] and the LHM [21].

We assume a virtual calibrated camera with image size of 640×480 pixels, focal length of 800 and principal point in the image center. We randomly generated 3D-to-2D correspondences, where 3D reference points were distributed into the interval $[-2,2] \times [-2,2] \times [4,8]$. We also added Gaussian noise to the 2D image coordinates. Finally, we chose the ground-truth translation $\mathbf{t}_{\text{true}}$ as the centroid of the 3D reference points and we randomly generated a ground truth rotation matrix $\mathbf{R}_{\text{true}}$. As a metric errors we used the same as in [6, 19, 33]. The absolute error is measured in degrees between the $\mathbf{R}_{\text{true}}$ and the estimated $\mathbf{R}$ as $\text{e}_{\text{rot}}(\text{deg}) = \max_{k=1}^{3}\{\text{acos}(\mathbf{r}_{\text{k,true}}^{\top} \cdot \mathbf{r}_k) \times 180/\pi\}$ where $\mathbf{r}_{\text{k,true}}$ and $\mathbf{r}_k$ are the k-th column of $\mathbf{R}_{\text{true}}$ and $\mathbf{R}$. The translation error is computed as $e_{\text{trans}}(\%) = \|\mathbf{t}_{\text{true}} - \mathbf{t}\|/\|\mathbf{t}\| \times 100$. All the plots discussed in this section were created by running 500 independent MATLAB simulations and report the average rotation and translation errors.

The first and second columns of Fig. 3 plot the accuracy for increasing number of correspondences, from $n = 10$ to 200. For each experiment we determine 10 different known isotropic Gaussian distributions, with standard deviations $\sigma = [1,...,10]$. The number of features corrupted with each distribution is equal to 10%.

The third and forth columns depict the errors for increasing amounts of noise. For each correspondence a uniform random σ is computed between 0 and the maximum image noise value, from 0 to 30. Assuming a constant number of correspondences $n = 100$.

Both experiments for planar and non-planar configurations, show that the proposed approach yields very accurate solutions, which are even better than the best state-of-the-art solution.

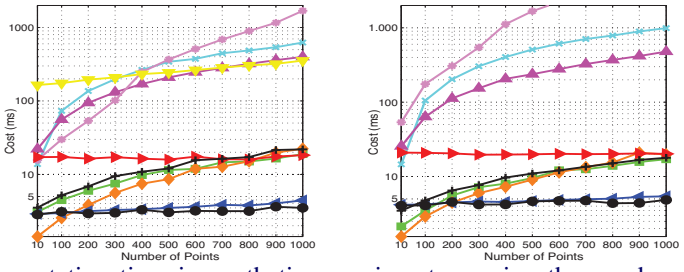


Figure 4: Computation time in synthetic experiments varying the number of correspondences. For the non-planar case (left) and the planar case (right) the color codes and line styles are the same as those used in Fig. 3.

Fig. 4 shows the computation time of all methods, for an increasing number of correspondences, from $n = 10$ to 1000 with known distributions with $\sigma = [1, ..., 10]$. This experiment was done on an Intel Core i7 CPU to 2.7Ghz and all methods are implemented in MATLAB. Note that our method, altough do not have a constant running time, is quite fast and has a linear running time respect to the number of correspondences.

4.2 Real images

We also tested our approach in real images. Feature points are obtained using SIFT [20] and correspondences are estimated using the repeteability measure [22], which yield around $200-400$ correspondences per image. The ground truth used to assess our approach is obtained by randomly generating 3,200 views using homographies with known camera poses. The camera pose for each view is generated with respect to an orthogonal reference view of a planar surface (similar to V_1 in Fig. 2a) by changing the camera orientations in pitch, yaw and roll angles. As the camera pose of test views moves away from the reference image, we

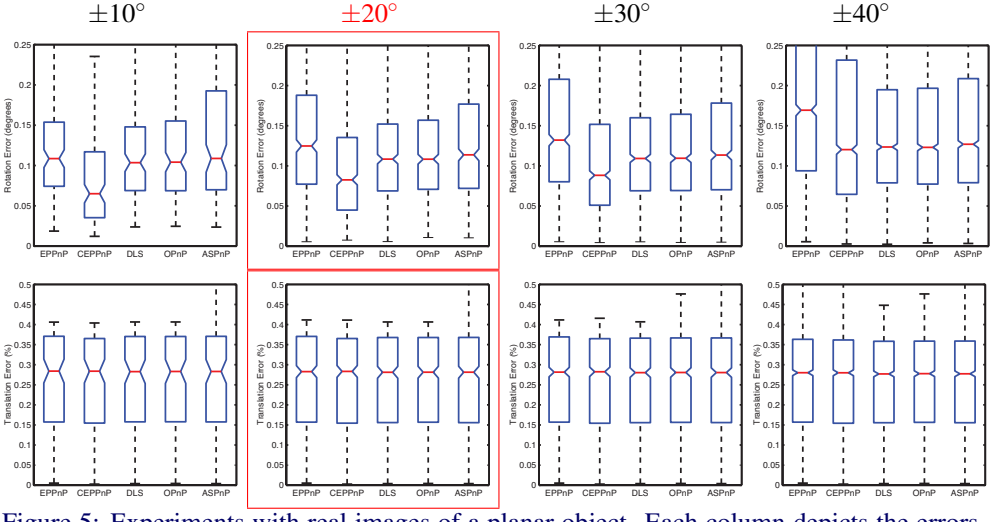


Figure 5: Experiments with real images of a planar object. Each column depicts the errors (median and quartiles) as the camera pose move away from the reference view, in rotation (first row) and translation (second row) terms. In red we remark the angular distance we have selected to generate the grid of reference images.

estimate the error with respect to the maximum absolute value of the pitch and yaw angles, from $\pm 10^\circ$ to $\pm 40^\circ$. The roll angle is not restricted since it does not affect directly to the perspective.

We compare CEPP*n*P against the methods showing the best results in the synthetic experiments of the planar case, namely DLS [11], ASP*n*P [34], OP*n*P [33] and EPP*n*P [5].

Fig. 5 shows that the pose results (in rotation) we obtain using CEPP*n*P are remarkably more consistent than all other approaches when using the feature uncertainties modeled by reference images close from the input image. This justifies the need of the three-step process we have described in Sect. 3.3, of distributing reference images for modeling the uncertainty all around the object at every 20° in pitch and yaw angles. In terms of translation error, all approaches yield almost the same accuracy. This is most probably due to the fact that we have only generated testing images by constraining the camera to be on the surface of an hemisphere on top of the object.

5 Conclusions

We have proposed a real-time and very accurate solution to the P*n*P problem that incorporates into its formulation the fact that in practice the 2D position of not all 2D features is estimated with the same accuracy. Our method approximates the Maximum Likelihood solution by minimizing an unconstrained Sampson error function. Furthermore we propose an effective strategy to model feature uncertainties on real images analyzing the sensitivity of feature detectors under viewpoints changes. Finally, we show that our approach outperforms the accuracy of state-of-the-art methods in both, synthetic and real experiments. As a future work we plan to transfer the 2D feature uncertainties to the 3D model, in order to make unnecessary to have a set of reference images with different 2D uncertainties.

Acknowledgement

This work has been partially funded by Spanish government under projects DPI2011-27510, IPT-2012-0630-020000, IPT-2011-1015-430000 and CICYT grant TIN2012-39203; by the EU project ARCAS FP7-ICT-2011-28761; and by the ERA-Net Chistera project ViSen PCIN-2013-047.

References

[1] Adnan Ansar and Konstantinos Daniilidis. Linear pose estimation from points or lines. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 25(5):578–589, 2003.

[2] Wojciech Chojnacki, Michael J. Brooks, Anton Van Den Hengel, and Darren Gawley. On the fitting of surfaces to data with covariances. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(11):1294–1303, 2000.

[3] Wojciech Chojnacki, Michael J Brooks, Anton van den Hengel, and Darren Gawley. FNS,CFNS and HEIV: A unifying approach. *Journal of Mathematical Imaging and Vision*, 23(2):175–183, 2005.

[4] Daniel DeMenthon and Larry S Davis. Exact and approximate solutions of the perspective-three-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 14(11):1100–1105, 1992.

[5] Luis Ferraz, Xavier Binefa, and Francesc Moreno-Noguer. Very fast solution to the PnP problem with algebraic outlier rejection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[6] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. of the ACM*, 24(6):381–395, 1981.

[7] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 25(8):930–943, 2003.

[8] Valeria Garro, Fabio Crosilla, and Andrea Fusiello. Solving the PnP problem with anisotropic orthogonal procrustes analysis. In *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPTV*, pages 262–269, 2012.

[9] Johann August Grunert. Das pothenotische problem in erweiterter gestalt nebst über seine anwendungen in geodäsie. *In Grunerts Archiv für Mathematik und Physik*, 1841.

[10] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge Univ Press, 2000.

[11] Joel A Hesch and Stergios I Roumeliotis. A direct least-squares (DLS) method for PnP. In *IEEE International Conference on Computer Vision (ICCV)*, pages 383–390, 2011.

[12] Kenichi Kanatani. Renormalization for unbiased estimation. In *IEEE International Conference on Computer Vision (ICCV)*, pages 599–606, 1993.

[13] Kenichi Kanatani. Optimization techniques for geometric estimation: Beyond minimization. In *Structural, Syntactic, and Statistical Pattern Recognition*, pages 11–30. 2012.

[14] Kenichi Kanatani and Yasuyuki Sugaya. High accuracy fundamental matrix computation and its performance evaluation. In *British Machine Vision Conference (BMVC)*, pages 217–226, 2006.

[15] Kenichi Kanatani, Ali Al-Sharadqah, Nikolai Chernov, and Yasuyuki Sugaya. Renormalization returns: Hyper-renormalization and its applications. In *European Conference on Computer Vision (ECCV)*, pages 384–397. 2012.

[16] Laurent Kneip, Davide Scaramuzza, and Roland Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2969–2976, 2011.

[17] Yoram Leedan and Peter Meer. Heteroscedastic regression in computer vision: Problems with bilinear constraint. *International Journal on Computer Vision (IJCV)*, 37: 127–150, 2000.

[18] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. EPnP: An accurate O(n) solution to the PnP problem. *International Journal on Computer Vision (IJCV)*, 81(2): 155–166, 2009.

[19] Shiqi Li, Chi Xu, and Ming Xie. A robust O(n) solution to the perspective-n-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 34 (7):1444–1450, 2012.

[20] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal on Computer Vision (IJCV)*, 60(2):91–110, 2004.

[21] C-P Lu, Gregory D Hager, and Eric Mjolsness. Fast and globally convergent pose estimation from video images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(6):610–622, 2000.

[22] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and Luc Van Gool. A comparison of affine region detectors. *International Journal on Computer Vision (IJCV)*, 65:43–72, 2005.

[23] Francesc Moreno-Noguer, Vincent Lepetit, and Pascal Fua. Accurate non-iterative O(n) solution to the PnP problem. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1–8, 2007.

[24] Francesc Moreno-Noguer, Vincent Lepetit, and Pascal Fua. Pose priors for simultaneously solving alignment and correspondence. In *European Conference on Computer Vision (ECCV)*, volume 2, pages 405–418, 2008.

[25] Carl Olsson, Fredrik Kahl, and Magnus Oskarsson. Branch-and-bound methods for euclidean registration problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31(5):783–794, 2009.

[26] Adrian Penate-Sanchez, Juan Andrade-Cetto, and Francesc Moreno-Noguer. Exhaustive linearization for robust camera pose and focal length estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(10):2387–2400, 2013.

[27] Long Quan and Zhongdan Lan. Linear n-point camera pose determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 21(8):774–780, 1999.

[28] Jordi Sánchez-Riera, Jonas Ostlund, Pascal Fua, and Francesc Moreno-Noguer. Simultaneous pose, correspondence and non-rigid shape. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1189–1196, 2010.

[29] Peter H Schönemann and Robert M Carroll. Fitting one matrix to another under choice of a central dilation and a rigid motion. *Psychometrika*, 35(2):245–255, 1970.

[30] Gerald Schweighofer and Axel Pinz. Globally optimal O(n) solution to the PnP problem for general camera models. In *British Machine Vision Conference (BMVC)*, pages 1–10, 2008.

[31] Eduard Serradell, Mustafa Özuysal, Vincent Lepetit, Pascal Fua, and Francesc Moreno-Noguer. Combining geometric and appearance priors for robust homography estimation. In *European Conference on Computer Vision (ECCV)*, pages 58–72, September 2010.

[32] Bill Triggs. Camera pose and calibration from 4 or 5 known 3d points. In *IEEE International Conference on Computer Vision (ICCV)*, volume 1, pages 278–284, 1999.

[33] Yinqiang Zheng, Yubin Kuang, Shigeki Sugimoto, Kalle Aström, and Masatoshi Okutomi. Revisiting the PnP problem: A fast, general and optimal solution. In *IEEE International Conference on Computer Vision (ICCV)*, pages 4321–4328, 2013.

[34] Yinqiang Zheng, Shigeki Sugimoto, and Masatoshi Okutomi. ASPnP: An accurate and scalable solution to the perspective-n-point problem. *Trans. on Information and Systems*, 96(7):1525–1535, 2013.