Robust Point Correspondence by Concave Minimization

João Maciel[*] and João Costeira
Instituto de Sistemas e Robótica – Instituto Superior Técnico
Av. Rovisco Pais, 1049-001 Lisboa Codex, PORTUGAL
{maciel,jpc}@isr.ist.utl.pt

Abstract

We propose a new methodology for reliably solving the correspondence problem between points of two or more images. This is a key step in most problems of Computer Vision and, so far, no general method exists to solve it.

Our methodology is able to handle most of the commonly used assumptions in a unique formulation, independent of the domain of application and type of features. It performs correspondence and outlier rejection in a single step, and achieves global optimality with feasible computation. Feature selection and correspondence are first formulated as an integer optimization problem. This is a blunt formulation, which considers the whole combinatorial space of possible point selections and correspondences. To find its global optimal solution we build a concave objective function and relax the search domain into its convex-hull. The special structure of this extended problem assures its equivalence to the original one, but it can be optimally solved by efficient algorithms that avoid combinatorial search.

1 Introduction

Estimating feature correspondences between two or more images is a long standing fundamental problem in Computer Vision. Most methods for 3D reconstruction, image alignment, object recognition or classification and camera self-calibration start by assuming that image feature points were extracted and put to correspondence. This is, therefore, a key problem and, so far, no general reliable method exists to solve it.

We propose a new methodology for reliably solving the point correspondence problem. There are three main difficulties associated with the problem. First, it is ill posed, and there are no general constraints to reduce its ambiguity. Second, it suffers from high complexity due to the huge dimensionality of the combinatorial search space. Finally the existence of outliers must be considered, since features can be missing or added through a sequence of images, due to occlusions and errors of the feature extraction procedure.

1.1 Previous Work

In order to deal with the ambiguity of the correspondence problem, all methods must impose domain-specific assumptions. These assumptions define the objective function — criterion — and the constraints of the optimization problem that is actually solved.

[*] Supported by *Subprograma Ciência e Tecnologia, 2º Quadro Comunitário de Apoio*

The most commonly used criterion is image correlation [13, 6], which reflects the assumption of image similarity. Another usual choice is proximity assumption [6]. The most general assumption is 3D scene rigidity [1, 12]. The smoothness of the disparity field is also used to disambiguate between multiple solutions [12, 9]. Examples of constraints are the order [9, 11], epipolar constraint [9, 11], and unicity [3]. Finally, the large diversity of current methods comes mainly from the many different optimization algorithms. Dynamic programming [9], graph search [11], and convex minimization [6] guaranty optimality. Non-optimal approaches include greedy algorithms, simulated annealing, relaxation [3], alternation of optimization and projection on the constraints [1] and randomized search [12, 13].

Most methods use domain specific formulations that set bounds on their range of applications and, some times, create implicit unwanted constraints. For instance, the method of [9] uses dynamic programming, which intrinsically assumes the order constraint. Furthermore, algorithms that do not consider the existence of outliers have poor performances when, for example, occlusions occur. Finally, most methods fail to guaranty global optimality. For example, [13] deals with outliers but depends on an iterative scheme, which is not proved to converge to the global solution.

2 Contribution

Our methodology is generic, in the sense that it is able to handle most of the commonly used assumptions in a unique formulation. It performs correspondence and outlier rejection in a single step and achives global optimality with feasible computation. As far as we know, no other method in the literature meets these three requirements simultaneously.

We use a single universal representation of the correspondence problem, independently of the domain of application and type of features. Both problems of feature selection and correspondence are formulated as a single integer optimization problem that considers the whole space of possible point selections and correspondences. We find its global solution avoiding combinatorial search without having to impose additional assumptions. We do so by relaxing the discrete search domain into its convex-hull. The special structure of the constraints and objective function assure that the extension generates an equivalent problem that can be optimally solved by efficient algorithms. By default, one can use a generic optimization algorithm — e.g. *simplex* — but some combinations of constraints and cost functions allow the use of even more efficient algorithms.

3 Methodology

For the sake of simplicity, we consider the two image correspondence problem. Extension to full sequences is straightforward. In section 4.2 we show a three-image situation.

3.1 Correspondence as an optimization problem

Consider two sets of feature-points observed by a stereo pair or by a moving camera. Consider that some of these pairs are the projections of the same 3D points. We collect the features of the first and second images, in two stacks of row vectors, respectively $\mathbf{X}$ and $\mathbf{Y}$. Features can represent the image coordinates of feature-points or any image-related quantity like a local neighborhood of intensities. The type of information conveyed by the features has to be coherent with the criterion, but does not affect our formulation.

Using the previous definitions we can formulate the correspondence problem as an integer minimization problem $\mathbf{P}^* = \arg\min_{\mathbf{P}} J(\mathbf{X}, \mathbf{Y}, \mathbf{P})$, where $\mathbf{P}$ is a zero-one variable that selects and sorts some rows of $\mathbf{Y}$, putting them to correspondence with the rows of $\mathbf{X}$. To guaranty robustness in the presence of outliers, $\mathbf{P}$ must allow some features not to be corresponded, so it cannot be a simple permutation. To avoid combinatorial explosion, we must be able to extend the zero-one domain. This results on the minimization problem 1 where J is a scalar objective function.

Problem 1

$$\mathbf{P}^* = \arg\min_{\mathbf{P}} \quad J(\mathbf{X}, \mathbf{Y}, \mathbf{P}) \qquad \text{s.t.} \quad \mathbf{P} \in \mathcal{P}_{\mathrm{p}}(p_1, p_2)$$

In this problem, $\mathbf{P}$ is constrained to $\mathcal{P}_{\mathrm{p}}(p_1, p_2)$, the set of $p_1 \times p_2$ *partial permutation matrices* ($\mathrm{p_p}$-matrices). A $\mathrm{p_p}$-matrix is a permutation matrix to which some columns and rows of zeros were added. Each entry $\mathbf{P}_{i,j}$ when set to 1 indicates that features $\mathbf{X}_{i\cdot}$ (row i of $\mathbf{X}$) and $\mathbf{Y}_{j\cdot}$ (row j of $\mathbf{Y}$) are put to correspondence.

$\mathrm{p_p}$-matrices represent, at most, one correspondence for each feature, and allow some features not to be matched. If row $\mathbf{P}_{i\cdot}$ is a row of zeros then feature $\mathbf{X}_{i\cdot}$ is not matched. If column $\mathbf{P}_{\cdot j}$ is a column of zeros then feature $\mathbf{Y}_{j\cdot}$ is not matched. Both correspondence and outlier rejection are intrinsic to this formulation because each element of $\mathcal{P}_{\mathrm{p}}(p_1, p_2)$ permutes only a subset of all the features. The global optimal solution of problem 1 is the best among all possible combinations of samples and permutations.

We generalize the usual definition of $\mathrm{p_p}$-matrices to non-square matrices, saying that any $p_1 \times p_2$ real matrix $\mathbf{P}$ is a $\mathrm{p_p}$-matrix *iff* it complies with the following conditions:

$$\mathbf{P}_{i,j} \in \{0, 1\}, \quad \forall i = 1 \ldots p_1, \quad \forall j = 1 \ldots p_2 \tag{1}$$

$$\textstyle\sum_{i=1}^{p_1} \mathbf{P}_{i,j} \leq 1, \quad \forall j = 1 \ldots p_2 \tag{2}$$

$$\textstyle\sum_{j=1}^{p_2} \mathbf{P}_{i,j} \leq 1, \quad \forall i = 1 \ldots p_1 \tag{3}$$

To avoid the trivial solution $\mathbf{P}^* = \mathbf{0}$, we establish a fixed number of correspondences $p_{\mathrm{t}} \leq \min(p_1, p_2)$ by considering a slightly different set of matrices $\mathcal{P}_{\mathrm{p}}^{p_{\mathrm{t}}}(p_1, p_2)$. We call these *rank-$p_{\mathrm{t}}$ partial permutation matrices* (rank-p_{t} $\mathrm{p_p}$-matrices). Constraining the optimization problem to $\mathcal{P}_{\mathrm{p}}^{p_{\mathrm{t}}}(p_1, p_2)$ leads to a process of picking up just the best p_{t} correspondences. Like in most robust methods [13], p_{t} should be kept near the minimum number of features required by the assumed model or lower than the estimated number of inliers. The definitions and properties of $\mathcal{P}_{\mathrm{p}}^{p_{\mathrm{t}}}(p_1, p_2)$ and other useful sets of matrices can be found in [8].

3.2 Reformulation with a compact convex domain

Problem 1 is a bluntly posed — brute force — integer minimization problem. In general, there is no efficient way to optimally solve such type of problems. Nonetheless there is a related class of optimization problems for which there are efficient, optimal algorithms. Such a class can be defined as Problem 2.

Problem 2

$$\mathbf{P}^* = \arg\min_{\mathbf{Q}} \quad J_\epsilon(\mathbf{X}, \mathbf{QY}) \qquad \text{s.t.} \quad \mathbf{Q} \in \mathcal{DS}_{\mathrm{s}}(p_1, p_2)$$

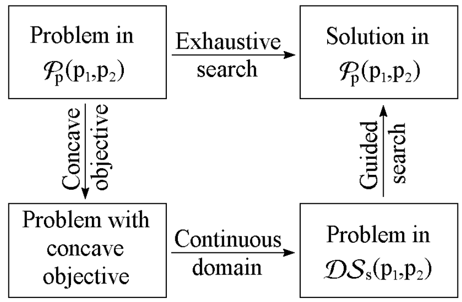

Figure 1: Efficient solution of the combinatorial problem.

where J_ϵ is a concave version of J — Equation 4 — and $\mathcal{DS}_\mathrm{s}(p_1, p_2)$ is the set of real $p_1 \times p_2$ *doubly sub-stochastic matrices*, the convex hull of $\mathcal{P}_\mathrm{p}(p_1, p_2)$.

Problems 1 and 2 can be made equivalent — same global optimal — by finding an adequate concave objective function J_ϵ. Also we must be sure that the vertices of $\mathcal{DS}_\mathrm{s}(p_1, p_2)$ are the elements of $\mathcal{P}_\mathrm{p}(p_1, p_2)$. Figure 1 summarizes the whole process. In short, the methodology is outlined as follows:

1. Extract points of interest and use their representations to build $\mathbf{X}$ and $\mathbf{Y}$.
2. Use $\mathbf{X}$ and $\mathbf{Y}$ to build the objective function J. Examples are Equations 10 and 13.
3. Use the procedure in Section 3.3 to produce an equivalent concave objective function J_ϵ — Equations 4 and 5.
4. Convert the desired convex constraints into the canonical form using Equations 7 and 8.
5. Solve the problem using a linear or concave programming algorithm.

Section 3.3 contains the proof for the equivalence of Problems 1 and 2. Extensions for the inclusion of other constraints can be found in [8].

3.3 Equivalence of the Problems 1 and 2.

Theorem 1 states the fundamental reason for the equivalence. [5] contains its proof.

Theorem 1 *A strictly concave function $J : \mathcal{C} \to \mathbb{R}$ attains its global minimum over a compact convex set $\mathcal{C} \subset \mathbb{R}^n$ at an extreme point of $\mathcal{C}$.*

The constraining set of a minimization problem with concave objective function can be changed to its convex-hull, provided that all the points in the original set are extreme points of the new compact set.

The problem now is how to find a concave function $J_\epsilon : \mathcal{DS}_\mathrm{s}(p_1, p_2) \to \mathbb{R}$ having the same values as J at every point of $\mathcal{P}_\mathrm{p}(p_1, p_2)$. Furthermore, we must be sure that the convex-hull of $\mathcal{P}_\mathrm{p}(p_1, p_2)$ is $\mathcal{DS}_\mathrm{s}(p_1, p_2)$, and that all $\mathrm{p_p}$-matrices are vertices of $\mathcal{DS}_\mathrm{s}(p_1, p_2)$, even in the presence of the rank-fixing constraint.

Consider Problem 1, where $J(\mathbf{q})$ is a class C^2 scalar function. Each entry of its Hessian is a continuous function $\mathbf{H}_{ij}(\mathbf{q})$. J can be changed to its concave version J_ϵ by

$$J_\epsilon(\mathbf{q}) = J(\mathbf{q}) + \sum_{i=1}^{n} \epsilon_i q_i^2 - \sum_{i=1}^{n} \epsilon_i q_i \quad (4)$$

Note that the constraints of Problem 1 include $q_i \in \{0,1\},\ \forall i$, so $J_\epsilon(\mathbf{q}) = J(\mathbf{q}),\ \forall \mathbf{q}$. On the other hand $\mathcal{P}_{\mathrm{p}}(p_1,p_2)$ is bounded by a hypercube $\mathcal{B} = \{\mathbf{q} \in \mathbb{R}^n : 0 \le q_i \le 1,\ \forall i\}$. All $\mathbf{H}_{ij}(\mathbf{q})$ are continuous functions so they are bounded for $\mathbf{q} \in \mathcal{B}$ — Weierstrass' theorem. This means that we can always choose a set of finite values ϵ_r, defined by

$$\epsilon_r \le -\frac{1}{2}\left(\max_{\mathbf{q}} \sum_{s=1,s\neq r}^{n} \left|\frac{\partial^2 J(q)}{\partial q_r \partial q_s}\right| - \min_{\mathbf{q}} \frac{\partial^2 J}{\partial q_r^2}\right) \tag{5}$$

which impose a negative stricly dominant diagonal to the Hessian of J_ϵ, that is to say, $|\mathbf{H}_{ii}| > \sum_{j=1,j\neq i}^{n} |\mathbf{H}_{ij}|\ ,\ \forall i$. A strictly diagonally dominant matrix having only negative elements on its diagonal is strictly negative definite [4], so these values of ϵ_r will guaranty that $J_\epsilon(\mathbf{q})$ is concave for $\mathbf{q} \in \mathcal{B}$ and, therefore, also for $\mathbf{q} \in \mathcal{DS}_{\mathrm{s}}(p_1,p_2)$.

Finally, note that problem 2 is constrained to the set of doubly sub-stochastic matrices, defined by conditions 2, 3 and 6

$$\mathbf{Q}_{i,j} \ge 0\ ,\ \forall i = 1 \dots p_1\ ,\ \forall j = 1 \dots p_2 \tag{6}$$

This set has the structure of a compact convex set in $\mathbb{R}^{p_1 \times p_2}$. Its extreme points are the elements of $\mathcal{P}_{\mathrm{p}}(p_1,p_2)$ — see [8]. This fact together with Theorem 1 proves that the continuous Problem 2 is equivalent to the original discrete Problem 1, since we're assuming that J_ϵ was conveniently made concave. If we use the $\mathcal{P}_{\mathrm{p}}^{p_{\mathrm{t}}}(p_1,p_2)$ set instead then its compact extension is $\mathcal{DS}_{\mathrm{s}}^{p_{\mathrm{t}}}(p_1,p_2)$, the set of *rank-*$p_{\mathrm{t}}$ *ds*$_{\mathrm{s}}$*-matrices* — see [8].

3.4 Constraints in canonical form

Most concave and linear programming algorithms assume that the problems have their constraints in canonical form. We now show how to put the constraints that define $\mathcal{DS}_{\mathrm{s}}(p_1,p_2)$ in canonical form, that is, how to state Problem 2 as

Problem 3

$$\begin{aligned} \mathbf{P}^* = \arg \min_{\mathbf{q}} \quad & J_\epsilon(\mathbf{X},\mathbf{Y},\mathbf{q}) \\ \text{s.t.} \quad & \mathbf{A}\mathbf{q} \le \mathbf{b}\ ,\ \mathbf{q} \ge \mathbf{0} \end{aligned}$$

where $\mathbf{A}_{[m\times n]}$ and $\mathbf{b}_{[m\times 1]}$ define the intersection of m left half-planes in $\mathbb{R}^n$.

The natural layout for our variables is a matrix $\mathbf{Q}$, so we use $\mathbf{q} = \mathrm{vec}(\mathbf{Q})$, where vec() stacks the columns of its operand into a column vector. Condition 2 is equivalent to $\mathbf{Q}.\mathbf{1}_{[p_2\times 1]} \le \mathbf{1}_{[p_1\times 1]}$. Applying the vec operator [7] to both sides of this inequality we obtain $\left(\mathbf{1}_{[1\times p_2]}^\top \otimes \mathbf{I}_{[p_1]}\right)\mathbf{q} \le \mathbf{1}_{[p_1\times 1]}$, where $\otimes$ is the Kronecker product, so set

$$\mathbf{A}_1 = \mathbf{1}_{[1\times p_2]}^\top \otimes \mathbf{I}_{[p_1]} \quad ; \quad \mathbf{b}_1 = \mathbf{1}_{[p_1\times 1]} \tag{7}$$

By the same token we express condition 3 as

$$\mathbf{A}_2 = \mathbf{I}_{[p_2]} \otimes \mathbf{1}_{[1\times p_1]}^\top \quad ; \quad \mathbf{b}_2 = \mathbf{1}_{[p_2\times 1]} \tag{8}$$

The intersection of conditions 2 and 3 results on the constraints of Problem 3 with

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \end{bmatrix} \quad ; \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \tag{9}$$

Similar conditions can be found for $\mathcal{DS}_{\mathrm{s}}^{p_{\mathrm{t}}}(p_1,p_2)$ and other useful sets.

3.5 Minimizing a linearly constrained concave function

To solve linear problems we use the the *lpSolve*[1] implementation of the *simplex* algorithm. For concave problems we use an extension of the exact method of [2]. Other exact methods provide more efficient procedures [5, 10] but this one is simple and easy to implement. Like the *simplex* algorithm, worst case complexity is factorial, but typically it visits only a small fraction of the vertices of the constraints. It performs surprisingly well in concave quadratic problems.

The method is based on a very simple iterative scheme. In each iteration the *next best* solution of a linear program is computed [14]. This can be accomplished by a few *simplex* pivoting steps. As iterations run, a sequence of ever-improving vertices of the constraining polytope is return, as well as a sequence of tighter and tighter bounds on the global minimum. Global optimality is tested by checking for coherence between the current best solution and the bounds.

Note that these are general-purpose algorithms, which solve the problems for any kind of constraints. If efficiency is an issue, then specialized algorithms should be used, wich have lower algorithmic complexity but will only work for a particular set of constraints. Examples are dynamic-programming [9] and graph-matching [11].

4 Experiments

In this section we will consider two of the most frequently used assumptions and insert them in the described framework. The resulting methods are tested in real images and their robustness is compared with benchmark algorithms.

4.1 Correlation matching by linear programming

Matching by correlation of image patches requires the solution of emerging ambiguities and outlier rejection. Our formulation solves both in a natural way. To use this criterion, features consist of image patches with N pixels centered around the previously segmented points of interest. Row i of $\mathbf{X}$ (and $\mathbf{Y}$) is the row vectorization of a patch around the ith feature-point of the first (and second) image. The sum of the correlation coefficients of the rows of $\mathbf{X}$ and $\mathbf{Y}$ is given by the matrix inner product of $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$, which are normalized to have zero mean and unit norm rows. So, the objective function is $J_1(\mathbf{Q}) = -\mathrm{tr}\left(\mathbf{Q}\hat{\mathbf{Y}}\hat{\mathbf{X}}^\top\right)$. Using algebraic properties of the trace operator [7]

$$\begin{aligned} J_1(\mathbf{q}) &= -\mathbf{c}_1^\top \mathbf{q} \\ \mathbf{c}_1 &= \mathrm{vec}\left(\hat{\mathbf{X}}\hat{\mathbf{Y}}^\top\right) \end{aligned} \tag{10}$$

which is linear in $\mathbf{q} = \mathrm{vec}(\mathbf{Q})$. Problem 3 can be solved by *simplex* algorithm.

4.1.1 Results

We compared the results of our method with those of two benchmark algorithms. The first algorithm solves the same problem with the same constraints using a greedy suboptimal

[1] written by M. Berkelaar and J. Dirks at Eindhoven U. of Technology

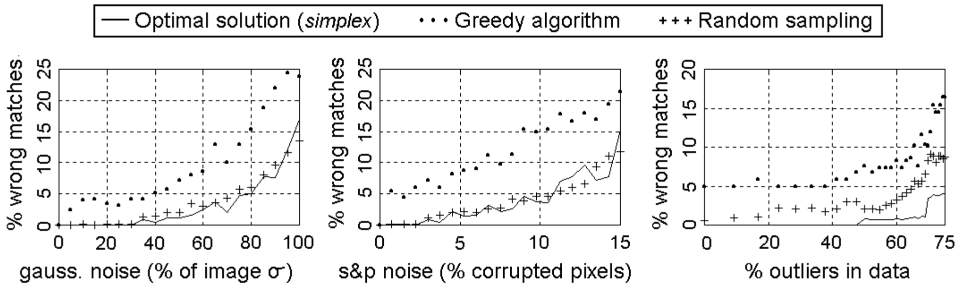

Figure 2: Average number of incorrect matches found in 40 trials for each noise level.

approach. The second benchmark improves the results of the first one by using a random sampling validation algorithm described in [13], that uses an extra rigidity assumption. This is known to achieve very reliable results. This algorithm randomly chooses sets of only a few of the feature pairs to estimate the Fundamental matrix, and keeps the solution with smallest median of the feature to epipolar distances. It then selects correspondences consistent with this Fundamental matrix. The number of iterations was set to 200.

We selected some image pairs with large disparity from the *Kitchen* sequence[2] and built three sets of corrupted data. The first set consisted of images with added zero-mean gaussian noise. In the second set, images were corrupted with salt-and-pepper noise. In the last set, outlier features — with randomly generated coordinates — were added to the extracted features, and all the images were corrupted with zero mean gaussian noise. We measured the number of incorrect matches returned by the three algorithms. The results are summarized on Figure 2.

4.1.2 Discussion

The greedy solution consistently produced higher number of mismatches, so we conclude that assuring optimality is a key factor on the reduction of the number of mismatches.

For the first two data sets, the random sampling algorithm returns the same percentage of mismatches of our method. When outliers are present, our method performs better. This is mainly because the validation procedure also rejects many good matches, which tends to raise the percentage of wrong matches.

The simultaneous rejection and correspondence of features is a reliable strategy, even in the presence of as much as 70% of outliers in data. The linear problems were solved in a fraction of a second by a *simplex* algorithm running on a 166MHz Pentium processor. In the case of 40 features plus 40 outliers, the cardinality of $\mathcal{P}_{\mathrm{p}}(p_1, p_2)$ is roughly 10^{70}. Exhaustive search would be impractical, while the *simplex* algorithm visits less than 300 solutions.

4.2 Epipolar search by quadratic concave minimization

Consider a trinocular system in generic configuration — focal points are not collinear — from which we known all Fundamental Matrices. Figure 3 shows the notation. Matrices

[2] Data was provided by the Modeling by Videotaping group in the Robotics Institute, CMU.

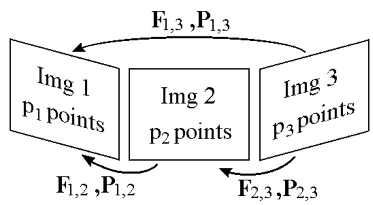

Figure 3: Notation for a trinocular system.

$\mathbf{P}_{1,2}$ and $\mathbf{P}_{1,3}$ are the variables of our problem. Each known Fundamental matrix $\mathbf{F}_{k,l}$ defines p_l epipolar lines $\mathcal{L}_{k,l}^{m}$ $, m = 1, \ldots, p_l$ on image k. A point on image k corresponding to the m-th point on image l must lie *close*[3] to $\mathcal{L}_{k,l}^{m}$. This defines a constraint that we represent by an indicator matrix $\mathbf{S}_{k,l}$. If entry (i, j) of $\mathbf{S}_{k,l}$ is set to 0, then entry (i, j) of $\mathbf{P}_{k,l}$ permanently set to 0. On the other hand, entry (i, j) of $\mathbf{S}_{k,l}$ is set to 1 if the i-th point on image k is close to $\mathcal{L}_{k,l}^{j}$. This means that entry (i, j) of $\mathbf{P}_{k,l}$ is a variable.

We represent these constraints implicitly with a *squeezed* set of variables $\mathbf{p}_{k,l}^{\mathrm{c}}$ of dimension $n_{k,l}$. These do not include the entries of $\mathbf{P}_{k,l}$ fixed to 0. We recover the full matrices through $\mathrm{vec}\left(\mathbf{P}_{k,l}\right) = \mathbf{B}_{k,l}\, \mathbf{p}_{k,l}^{\mathrm{c}}$, so the sub-stochastic constraints become

$$\left[\begin{array}{c} \mathbf{1}_{[1\times p_k]}^{\top} \otimes \mathbf{I}_{[p_l]} \\ \mathbf{I}_{[p_k]} \otimes \mathbf{1}_{[1\times p_l]}^{\top} \end{array} \right] \mathbf{B}_{k,l}\, \mathbf{p}_{k,l}^{\mathrm{c}} \leq \mathbf{1}_{[1\times n_{k,l}]} \tag{11}$$

We close the loop by estimating the *compound correspondence* $\hat{\mathbf{P}}_{2,3} = \mathbf{P}_{1,2}^{\top}\mathbf{P}_{1,3}$ and checking its coherence with $\mathbf{S}_{2,3}$. The objective function is

$$J_2 = \sum_{i=1}^{p_1}\sum_{j=1}^{p_2} \left(\mathbf{P}_{1,2} \odot \mathbf{D}_{1,2} + \mathbf{P}_{1,3} \odot \mathbf{D}_{1,3} + \hat{\mathbf{P}}_{2,3} \odot \mathbf{D}_{2,3} + \hat{\mathbf{P}}_{2,3} \odot \bar{\mathbf{S}}_{2,3} \right)_{i,j} \tag{12}$$

where $\odot$ is the elementwise product, $\bar{\mathbf{S}}_{2,3} = \left(\mathbf{1}_{[p_2\times p_3]} - \mathbf{S}_{2,3}\right)$, and $\mathbf{D}_{k,l}(i, j)$ is a matrix with the distances between the points $i = 1, \ldots, p_k$ of image k and the epipolar lines $\mathcal{L}_{k,l}^{j}$. The lack of coherence between $\hat{\mathbf{P}}_{2,3}$ and $\mathbf{S}_{2,3}$ is penalized by the last term. The other terms are the sum of all point-to-epipolar distances. These will disambiguate between different compatible solutions. By algebraic manipulation, we get the objective function

$$J_2\left(\mathbf{p}\right) = \mathbf{p}^{\top}\mathbf{J}_2\mathbf{p} + \mathbf{c}_2^{\top}\mathbf{p} \tag{13}$$

written in a complete vector of variables $\mathbf{p} = \left[\begin{array}{c} \mathbf{p}_{1,2}^{\mathrm{c}} \\ \mathbf{p}_{1,3}^{\mathrm{c}} \end{array} \right]$, and with

$$\begin{aligned} \mathbf{J}_2 &= \mathbf{B}_{1,2}^{\top}\left[\left(\mathbf{D}_{2,3} + \bar{\mathbf{S}}_{2,3}\right) \otimes \mathbf{I}_{[p_1]}\right]\mathbf{B}_{1,3} \\ \mathbf{c}_2 &= \left[\begin{array}{c} \mathbf{B}_{1,2}^{\top}\,\mathrm{vec}\left(\mathbf{D}_{1,2}\right) \\ \mathbf{B}_{1,3}^{\top}\,\mathrm{vec}\left(\mathbf{D}_{1,3}\right) \end{array} \right] \end{aligned}$$

Note that $\mathbf{J}_2$ is, in general, not concave, so a concave version J_ϵ must be computed using Equations 4 and 5, before the minimization algorithm is applied.

[3] define a distance threshold or choose a few from the nearest

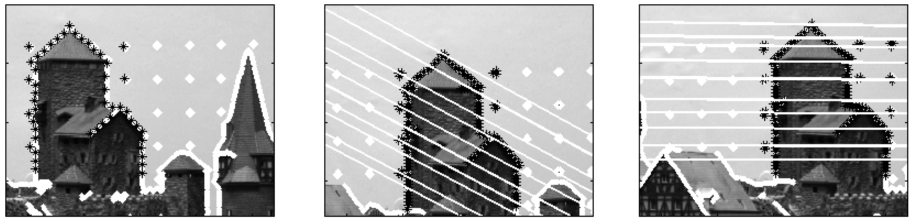

Figure 4: Three images from the *Castle* sequence, and some of the epipolar lines.

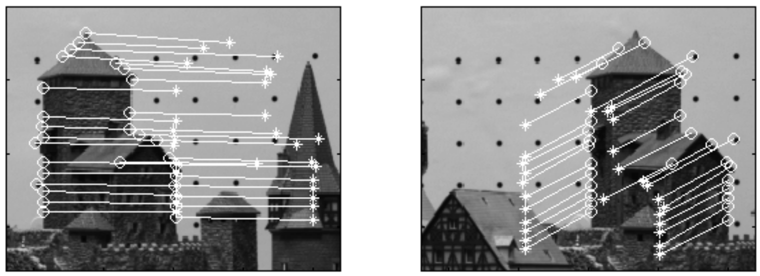

Figure 5: Graphical representation of the obtained $\mathbf{P}_{1,2}$ and $\mathbf{P}_{1,3}$.

4.2.1 Results

We applied the described method to the images of Figure 4, which are details taken from the *Castle* sequence[4]. The white lines are epipolar lines corresponding to a few points of the first image. The white dots are edge points from the Canny edge extractor. The feature points — crosses in black — come from inside a manually defined rectangular region of interest. They were automatically chosen from the set of edges by a bucketing procedure to guaranty a minimum distance between them. At the end we obtained 50 points from the first image and 130 points from each of the other images. Note that the second and third images contain, at least, 80 outliers, so the problem is solved in the presence of roughly 60% of outliers in the data. We fixed the number of computed correspondences by setting $p_{\mathrm{t}} = 30$, and obtained the correspondences in Figure 5.

4.2.2 Discussion

We detected 3 mismatches in $\mathbf{P}_{1,2}$ and 2 mismatches in $\mathbf{P}_{1,3}$, corresponding to 8% errors. The algorithm stopped after less than 100 iterations, when the bounds — Section 3.5 — were closer than a fixed threshold, so the depicted solution is not optimal. We set this threshold to admit only solutions without violations of the epipolar constraints, though mismatches can occur when more than one point is within the lozenge defined by the crossing of two epipolar lines. The mismatches occour because the first three terms of the objective function — Equation 12 — are unable to correctly disambiguate these situations, so the introduction of other assumptions would result on a better performance.

[4] Data was provided by the Calibrated Imaging Laboratory at CMU.

5 Conclusion

We have shown a methodology to solve the correspondence problem, which avoids unwanted assumptions by requiring their explicit statement. Furthermore it reliably handles outliers, even in situations where other robust methods fail.

The most important limitation of the methodology is the dimensionality of the optimization problems, specially when the objective functions are high-order polynomials. A practical way of minimizing this is the selection of a small number of reliable features in one of the images. Ongoing work is being conducted on the implementation of an efficient algorithm for high-order polynomial problems, and dealing with the assumption of rigidity under various camera models — see [8]. We plan to extend the methodology to dealing with extended sequences of images. Also we are working on reducing the dimensionality of the problem by integration of more than one assumption.

References

[1] R. Berthilsson and K. Astrom. Reconstruction of 3D-curves from 2D-images using affine methods for curves. In *Proc. CVPR*, 1997.

[2] A. Cabot and R. Francis. Solving certain nonconvex quadratic minimiation problems by ranking the extreme points. *Operations Research*, 18(1):82–86, Feb 1970.

[3] S. Gold, A. Rangarajan, C. Lu, S. Pappu, and E. Mjolsness. New algorithms for 2D and 3D point matching. *Pattern Recognition*, 31(8):1019–1031, 1998.

[4] R. Horn and C. Johnson. *Matrix Analysis*. Cambridge U. Press, 1985.

[5] R. Horst and P. Pardalos, editors. *Handbook of Global Optimization*. Kluwer, 1995.

[6] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. 7th International Joint Conference on AI*, 1981.

[7] H. Lutkepohl. *Handbook of Matrices*. Wiley, 1996.

[8] J. Maciel and J. Costeira. Stereo matching as a concave minimization problem. Technical Report VISLAB-TR 11/99, Instituto de Sistemas e Robótica, IST, 1999. http://www.isr.ist.utl.pt/˜maciel/tr1199.ps.gz.

[9] Y. Ohta and T. Kanade. Stereo by intra- and inter-scanline search using dynamic programming. *IEEE Trans. PAMI*, 7(2):139–154, March 1985.

[10] P. Pardalos and J. Rosen. Methods for global concave minimization: A bibliographic survey. *SIAM Review*, 28(3):367–379, Sep 1986.

[11] S. Roy and I. Cox. A maximum-flow formulation of the n-camera stereo correspondence problem. In *Proc. ICCV*, 1997.

[12] G. Sudhir, S. Banerjee, and A. Zisserman. Finding point correspondences in motion sequences preserving affine structure. *CVIU*, 68(2):237–246, November 1997.

[13] P. Torr. *Motion Segmentation and Outlier Detection*. PhD thesis, Dept. Engineering Science, U. Oxford, 1995.

[14] S. Wallace. Pivoting rules and redundancy schemes. *BIT*, 25:274–280, 1985.