

# Modelling compatibility coefficient distributions for probabilistic feature-labelling schemes

W.J. Christmas, J. Kittler and M. Petrou  
Vision, Speech and Signal Processing Group  
Department of Electronic and Electrical Engineering  
University of Surrey

email: [w.christmas,j.kittler,m.petrou@ee.surrey.ac.uk](mailto:w.christmas,j.kittler,m.petrou@ee.surrey.ac.uk)

www: <http://www.ee.surrey.ac.uk/>

## Abstract

In previous work, a probabilistic method for matching of 2-D geometric features has been described. For this method, a noise model is required for the compatibility coefficients, which has hitherto been specified heuristically. In this paper, the authors show how an improved noise model can be derived, using as a basis the noise model of the feature measurements. The new noise model is shown to provide better performance than the heuristic one. Using a non-iterative version of the matching algorithm, a good solution was found using the new model, which was not the case for the old one. When an iterative version was used, both models generated good solutions, but the new model resulted in faster convergence. Furthermore, no training is needed to generate covariances for the new model. A means of explicitly modelling small scaling errors is also given.

## 1 Introduction

Many authors have described methods for feature labelling based on probabilistic considerations, in which features extracted from a scene are matched with appropriate label features from some model. In most cases the methods exploit measurements in the form of relations between the features, and use an iterative or relaxation scheme to propagate the effects of the relational measurements throughout the network of features. One of the earliest such methods was that of Rosenfeld, Hummel and Zucker [8] (RHZ), which was essentially heuristic in nature. However, later workers were able to demonstrate that the RHZ scheme, and schemes closely related to it, have some useful mathematical properties. The work of Hummel and Zucker [4] introduced the notion of a consistent solution, and showed that the relaxation process could lead to such a solution. They were also able to show that their relaxation scheme could be expressed as an optimisation scheme, and that stable, convergent solutions could be obtained. What all of these methods have in common is the incorporation of the relational information in the form of compatibility coefficients, indicating to what extent each of the pairs of features extracted from the scene was compatible with each possible pair of label features in the model. However, with these

methods the determination of the compatibility coefficients does not follow automatically from the derivation of the algorithm, and some other method of determining them has to be found (*e.g.* [7]).

A slightly different approach was followed by Kittler and Hancock[6] and Christmas, Kittler and Petrou [2, 1], who used a Bayesian approach to look for solutions based on the Maximum A Posteriori (MAP) probabilities of the possible labellings. These schemes also generally relied on a relaxation algorithm to provide an unambiguous solution (with the exception of [2]). They also led to a slightly different form of update rule, and, more importantly, in contrast to most of the other probabilistic methods, showed directly that the compatibility coefficients should be calculated from the density functions of the relational measurements. In the versions of the method concerned with geometric feature matching [2, 1] (the CKP method), a fairly simple noise model is used to generate the density functions for the relational measurements. In particular it required the estimation of several constant parameters, without indicating why they should be constant, or how they should be calculated other than by estimating them through a training process. In this paper we provide a more rigorous noise model, which is based on more realistic assumptions, and which automatically generates the required parameters without requiring a training session. Although the method presented here also in turn requires some parameters to be estimated, there are fewer of them (typically one or two), and a rationale for their estimation is provided.

In section 2 we briefly describe the CKP labelling algorithm of [1], and highlight the limitations of the noise model used. In section 3 we indicate how to generate the distributions for the binary measurements from the unary measurement errors, and in section 4 we analyse the sources of the measurement accuracy. Some results are presented in section 5 and the method is reviewed in section 6.

## 2 The CKP labelling algorithm

In this section we discuss briefly the derivation of the version of the CKP algorithm in [1], indicating how it is related to the traditional relaxation scheme of [4].

We have a set of  $N$  objects, or features,  $\{a_1, \dots, a_N\}$ , extracted from the scene, each of which has to be labelled with a label from the model label set  $\Omega = \{\omega_0, \dots, \omega_M\}$ . Note that there are  $M$  model labels, corresponding to  $M$  physical model features, plus the null label  $\omega_0$  which is used to provide a notional label for scene features for which no other label is appropriate. We denote by  $a_i \leftarrow \omega_\alpha$  the labelling of object  $a_i$  with label  $\omega_\alpha$ .

Each scene feature  $a_i$  has an associated set of  $m_1$  unary measurements:

$$\mathbf{x}_i = \{x_i^{(1)}, \dots, x_i^{(m_1)}\}$$

and for each pair of scene measurements  $a_i$  and  $a_j$  there is a set of  $m_2$  binary measurements:

$$A_{ij} = \{A_{ij}^{(1)}, \dots, A_{ij}^{(m_2)}\}$$

There are corresponding sets of unary and binary measurements,  $\check{\mathbf{x}}_\alpha$  and  $\check{A}_{\alpha\beta}$  relating to the model features. We define the set of indices  $N_i$  by:

$$N_i \equiv \{1, \dots, i-1, i+1, \dots, N\}$$

The CKP MAP rule is then stated as: the most appropriate label  $\omega_\tau$  for object  $a_i$  is given by:

$$\omega_\tau = \arg \left[ \max_{\omega_\lambda \in \Omega} P(a_i \leftarrow \omega_\lambda | \{\mathbf{x}_j \forall j\}, \{A_{ij} \forall j \in N_i\}) \right] \quad (1)$$

By repeated application of Bayes's rule, and after making some assumptions, the conditional probability term on the right-hand side is expanded as follows:

$$P(a_i \leftarrow \omega_\lambda | \{\mathbf{x}_j \forall j\}, \{A_{ij} \forall j \in N_i\}) = \frac{P(a_i \leftarrow \omega_\lambda | \mathbf{x}_i) Q(a_i \leftarrow \omega_\lambda)}{\sum_{\omega_\alpha \in \Omega} P(a_i \leftarrow \omega_\alpha | \mathbf{x}_i) Q(a_i \leftarrow \omega_\alpha)} \quad (2)$$

where

$$Q(a_i \leftarrow \omega_\alpha) = \prod_{j \in N_i} \sum_{\omega_\beta \in \Omega} P(a_j \leftarrow \omega_\beta | \mathbf{x}_j) p(A_{ij} | a_j \leftarrow \omega_\beta, a_i \leftarrow \omega_\alpha) \quad (3)$$

We can write the relaxation rule of RHZ as

$$P^{(n+1)}(a_i \leftarrow \omega_\lambda) = \frac{P^{(n)}(a_i \leftarrow \omega_\lambda) Q^{(n)}(a_i \leftarrow \omega_\lambda)}{\sum_{\omega_\alpha \in \Omega} P^{(n)}(a_i \leftarrow \omega_\alpha) Q^{(n)}(a_i \leftarrow \omega_\alpha)} \quad (4)$$

where

$$Q^{(n)}(a_i \leftarrow \omega_\alpha) = \sum_{j \in N_i} \sum_{\omega_\beta \in \Omega} P^{(n)}(a_j \leftarrow \omega_\beta) r_{ij}^{\alpha\beta} \quad (5)$$

and where  $r_{ij}^{\alpha\beta}$  denotes the compatibility between features  $i$  and  $j$  in the scene and labels  $\alpha$  and  $\beta$  in the model. Having noticed the similarity between the form of expressions (2) and (3) and the relaxation rule of RHZ, the authors of the CKP rule then created a similar iterative rule from (2) and (3):

$$P^{(n+1)}(a_i \leftarrow \omega_\lambda) = \frac{P^{(n)}(a_i \leftarrow \omega_\lambda) Q^{(n)}(a_i \leftarrow \omega_\lambda)}{\sum_{\omega_\alpha \in \Omega} P^{(n)}(a_i \leftarrow \omega_\alpha) Q^{(n)}(a_i \leftarrow \omega_\alpha)} \quad (6)$$

where

$$Q^{(n)}(a_i \leftarrow \omega_\alpha) = \prod_{j \in N_i} \sum_{\omega_\beta \in \Omega} P^{(n)}(a_j \leftarrow \omega_\beta) p(A_{ij} | a_j \leftarrow \omega_\beta, a_i \leftarrow \omega_\alpha) \quad (7)$$

Prior labelling probabilities are used to initialise the iterations. Once the iterations have converged (see [9]), the MAP rule is then used to select the most appropriate labelling. Although the authors offer no proof, the process of iteration in practice ensures that this labelling is consistent.

We note that the only significant difference between the two rules is the replacement of the outer summation in the RHZ support function by a product in the CKP support function. Thus in the CKP support function (7), the role of the compatibility coefficients is clearly played by the probability density functions for the binary measurements.

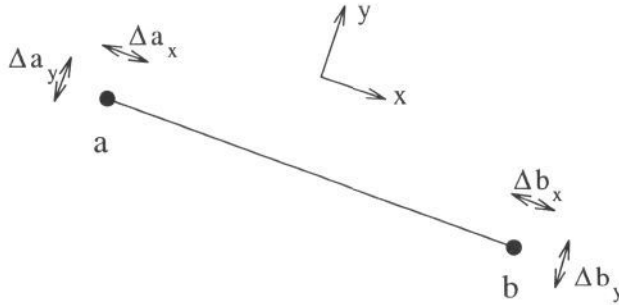
### 3 Evaluation of the probability density functions

In [1] it was indicated that the compatibility coefficients can be found by evaluating the joint density functions of the binary measurement errors. These were assumed to be Gaussian, with mean given by the model measurements  $\bar{A}_{\alpha\beta}$ :

$$p(A_{ij} | a_i \leftarrow \omega_\alpha, a_j \leftarrow \omega_\beta) = \mathcal{N}_{A_{ij}}(\bar{A}_{\alpha\beta}, \Sigma_2) \quad (8)$$

where  $\Sigma_2$  is the covariance matrix for the binary measurements  $A_{ij}$ . In an earlier, related paper [5], the authors also assume that the binary measurements are independent, and imply that the diagonal covariance matrix terms are constant over all the measurement pairs. No indication was given as to how the covariance matrix  $\Sigma_2$  should be estimated, other than by a training process. In reality, neither of the above assumptions holds in general. Here we will assume instead that the errors of the *unary* measurements only are independent. In this section we show how the errors of the binary measurements are derived from those of the unary measurements, and hence derive the covariances for the individual feature pairs as functions of the unary measurement variances. The analysis holds provided the errors are sufficiently small that the binary errors can be considered to be linear combinations of the unary ones. The analysis is general, and does not rely on the assumption of a Gaussian distribution.

We consider here an application in which the features are straight line segments, and in which the unknown transformation between scene and model spaces is Euclidean (*i.e.* the only unknowns are the relative translation and rotation). Each segment is characterised by the four measurements which define its two endpoints  $a$  and  $b$ . Using a local coordinate system aligned with the segment, we define errors for each measurement:



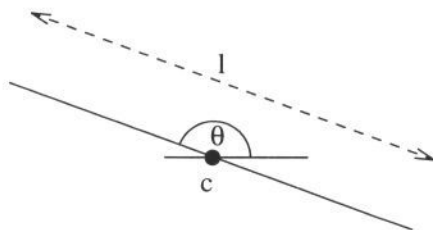
We make the following simplifying assumptions:

1. The error in the position of a segment endpoint can be characterised by two independent distributions, one collinear and one perpendicular to the segment; the respective variances are denoted as  $\sigma_{xx}$  and  $\sigma_{yy}$ .
2. The errors of the two endpoints of a segment are independent.
3. The errors of a segment are independent of those of other segments.

The last two assumptions do not always hold; for example two neighbouring segments could have a common endpoint. However such cases are the the exception rather than the rule.

To simplify the notation, we also assume in this section that the distributions of the endpoint errors are the same for all of the segments. It is a minor additional complication to deal with cases in which this is not true.

In order to derive the binary measurements, it is more convenient to represent the segment parameters by the position of its centre point  $(c_x, c_y)$  and its length  $l$  and global orientation  $\theta$ :



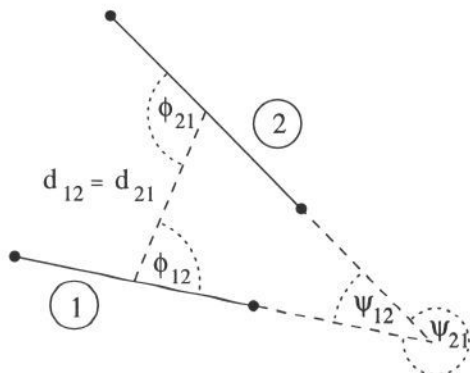
Thus, assuming that the measurement errors are small compared to the segment length:

$$\begin{aligned}\Delta c_x &= \frac{\Delta b_x + \Delta a_x}{2} \\ \Delta c_y &= \frac{\Delta b_y + \Delta a_y}{2} \\ \Delta \theta &= \frac{\Delta b_y - \Delta a_y}{l} \\ \Delta l &= \Delta b_x - \Delta a_x\end{aligned}$$

The corresponding variances are then given by:

$$\begin{aligned}\sigma_{c_x c_x} &= \frac{\sigma_{xx}}{2} \\ \sigma_{c_y c_y} &= \frac{\sigma_{yy}}{2} \\ \sigma_{\theta \theta} &= \frac{2\sigma_{yy}}{l^2} \\ \sigma_{ll} &= 2\sigma_{xx} \\ \sigma_{c_x c_y} &= \sigma_{c_x \theta} = \sigma_{c_x l} = \sigma_{c_y \theta} = \sigma_{c_y l} = \sigma_{\theta l} = 0\end{aligned}$$

The segment length  $l$  turns out in practice not to generate useful relations, but is needed in estimating the relation variances. We use the remaining three unary measurements to define three binary measurements for a pair of segments; these are the angle  $\psi$  between the segments, and the position of the centre of one segment with respect to that of the other expressed in local polar coordinates  $d, \phi$ :



From these binary measurements the errors can be calculated as:

$$\begin{aligned}\Delta d_{12} &= -\cos \phi_{12} \Delta c_{1x} - \sin \phi_{12} \Delta c_{1y} - \cos \phi_{21} \Delta c_{2x} - \sin \phi_{21} \Delta c_{2y} \\ \Delta \phi_{12} &= \frac{1}{d_{12}} (\sin \phi_{12} \Delta c_{1x} - \cos \phi_{12} \Delta c_{1y} + \sin \phi_{21} \Delta c_{2x} - \cos \phi_{21} \Delta c_{2y}) - \Delta \theta_1 \\ \Delta \psi_{12} &= \Delta \theta_2 - \Delta \theta_1\end{aligned}$$

The variances and covariances are therefore given by:

$$\begin{aligned}\sigma_{d_{12}d_{12}} &= (\cos^2 \phi_{12} + \cos^2 \phi_{21}) \frac{\sigma_{xx}}{2} + (\sin^2 \phi_{12} + \sin^2 \phi_{21}) \frac{\sigma_{yy}}{2} \\ \sigma_{\phi_{12}\phi_{12}} &= \frac{1}{2d_{12}^2} [(\sin^2 \phi_{12} + \sin^2 \phi_{21}) \sigma_{xx} + (\cos^2 \phi_{12} + \cos^2 \phi_{21}) \sigma_{yy}] + \frac{2\sigma_{yy}}{l_1^2} \\ \sigma_{\psi_{12}\psi_{12}} &= 2 \left( \frac{1}{l_1^2} + \frac{1}{l_2^2} \right) \sigma_{yy} \\ \sigma_{d_{12}\phi_{12}} &= \frac{1}{4d_{12}} (\sin 2\phi_{12} + \sin 2\phi_{21}) (\sigma_{yy} - \sigma_{xx}) \\ \sigma_{d_{12}\psi_{12}} &= 0 \\ \sigma_{\phi_{12}\psi_{12}} &= \frac{2\sigma_{yy}}{l_1^2}\end{aligned}$$

Thus if for example the unary measurement errors were Gaussian, the compatibility coefficients would be given by (8), in which the components of  $\Sigma_2$  are given by the above. It is apparent from these expressions that, since several of the covariance matrix terms are functions of the binary relations themselves, the earlier assumption that they are constant was an oversimplification. This is demonstrated quantitatively later (section 5.1).

## 4 Estimating the measurement errors in practice

The method described above has been evaluated on a number of different applications, in which the features were line segments both directed and undirected. Point features have also been used. In practice, three distinct sources of positional errors in the feature measurements were encountered:

1. Errors due to noise and other impairments in the image.
2. Uncertainties in the collinear location of the segment end points due to line breakage. Line breakage occurs for different reasons in different applications:
  - (a) although the features were genuinely straight lines, image noise and feature occlusion sometimes caused the feature detection process to break the segment into smaller ones; also occlusions by the image boundary often occurred;
  - (b) the underlying features were curves that were approximated by sets of straight line segments in both scene and model: here the position of the segment endpoints along the curve is often arbitrary, and does not in general correspond between scene and model;
  - (c) long line segments were deliberately broken in order to localise the features for the matching process.

3. Uncertainties in the transformation between scene and model frames of reference. Although the applications had unknown transformations that were nominally Euclidean (or translational), in practice *small* scaling errors or other distortions were also often encountered.

The actual noise parameters for errors of type 1 can be found from an analysis of the processes that generate the segments from the original image; such an analysis can be found for example in [3], which provides a means for estimating  $\sigma_{xx}$  and  $\sigma_{yy}$  directly.

Errors of type 2 essentially affect only  $\sigma_{xx}$ . In practice, the effect of line breakage is that the location of segment end points in a direction collinear with the segment is poorly defined, resulting in a large value of  $\sigma_{xx}$  that swamps the effect of errors of type 1. Accordingly  $\sigma_{xx}$  is typically set to a value of around  $(l/2)^2$ .

Errors of type 3 are harder to accommodate. Invariance to scaling errors cannot be explicitly handled using unary and binary measurements alone without throwing away much of the structural information. In practice however, small amounts of scale distortion may be accommodated by adding an extra error term to the distance relations that is proportional to the distance itself. Thus, if the scaling error is a random variable with variance  $\sigma_{ss}$ , the expression for  $\sigma_{d_{12}d_{12}}$  becomes:

$$\sigma_{d_{12}d_{12}} = (\cos^2 \phi_{12} + \cos^2 \phi_{21}) \frac{\sigma_{xx}}{2} + (\sin^2 \phi_{12} + \sin^2 \phi_{21}) \frac{\sigma_{yy}}{2} + d_{12} \sigma_{ss}$$

We note, however, that this solution suffers from the drawback that the scaling error is being modelled as a *separate* random variable for each measurement. This is clearly not true, and means that we are ignoring the fact that the scaling error is a single constant. The effect is to give emphasis to binary measurements of nearby nodes in contrast to those of more distant nodes.

There are other systematic errors which we may also wish to model in a similar way. Consider for example an image of some structure for which we have a 3-D model: we wish to match corresponding features from the image and model. If the position of the camera is known approximately, the model can be projected into the image plane, with some error. Using knowledge of the projection process, extra terms can be added to the variances in a similar manner to that for the scaling error. For stereo matching, the epipolar constraint can similarly be used.

## 5 Comparison of the new method with the use of fixed variances

We now demonstrate the two noise models on a typical matching problem, in which road segments are used as features. The image in this case is an aerial image of the ground; the roads were extracted from this image, and the resulting pixel strings were approximated by a set of straight line segments. A digital map covering a much larger area was used to generate a corresponding set of model segments. The image was of unknown orientation, but the scale was approximately that of the map. Short segments (less than 8 pixels long) were omitted from image and model feature sets.

## 5.1 Using the new noise model

As the signal-to-noise ratio of the image was good, the main positional errors in the image features were:

- the intensity ridge detection process, which was accurate to between 1/2 and 1 pixel, and
- the approximation of the pixel strings by line segments, for which the variance of the fitted segments was  $0.1 \text{ pixel}^2$

On this basis, we put  $\sigma_{yy} = 1 \text{ pixel}^2$ , and from the arguments of the previous section we put  $\sigma_{xx} = (l/2)^2$ . We used a Gaussian noise model (for ease of computation).

The results of the matching are shown in fig. 1. An iterative algorithm was used, converging to a stable solution after two iterations. Matched features in image and map are shown in white; unmatched map features are grey. The relative position and orientation of the image with respect to the map was also computed, using a least-mean-square fitting of the matched segments; this is indicated by the (black) pointer in image and map. From the figure we can see that all of the image features were correctly labelled on the map, and the correct orientation and position were found. When a non-iterative version of the algorithm was used, two of the scene features were labelled with the null label, and the remaining labellings were correct.

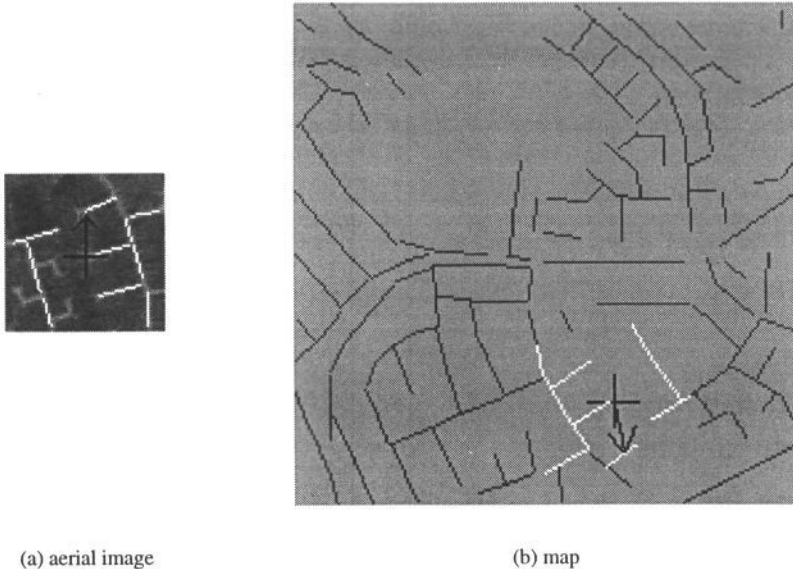


Figure 1: Matching of road segments

Some comments are in order on the range of covariance matrix values generated by this method. The means and standard deviations of the diagonal terms, computed over all of the image relations, are:

	mean	s.d.
$\sigma_{dd}$ (pixel <sup>2</sup> )	48.4	37.6
$\sigma_{\phi\phi}$ (rad <sup>2</sup> )	0.13	0.21
$\sigma_{\psi\psi}$ (rad <sup>2</sup> )	0.050	0.014

As noted earlier, two of these terms,  $\sigma_{dd}$  and  $\sigma_{\phi\phi}$ , are dependent on the relational measurements themselves, and so we would expect them to vary considerably. On the other hand the third term,  $\sigma_{\psi\psi}$ , is dependent only on the feature lengths, which in our example are relatively constant. This is reflected in the ratio of the standard deviation to the mean for each term.

Of the off-diagonal terms,  $\sigma_{d\phi}$  varies widely – the correlation coefficient varies over almost the entire range of  $[-1 \dots +1]$ . The other non-zero term,  $\sigma_{\phi\psi}$ , is dependent only on the feature lengths, and thus is again relatively constant.

## 5.2 Using fixed variances

With the earlier method [2, 1], the values for the variances have first to be estimated. For this 19 images were used, for which the correct matches were known. The position and rotation errors were measured, and from this the values for the variances were estimated:

$\sigma_{dd}$	65.6 pixel <sup>2</sup>
$\sigma_{\phi\phi}$	0.13 rad <sup>2</sup>
$\sigma_{\psi\psi}$	0.060 rad <sup>2</sup>

We note that these values are all within one standard deviation of the mean values generated by the new noise model.

The same image and model as before (section 5.1) were used. With the non-iterative matching algorithm, only one image feature was correctly labelled, null labels being selected for all of the remaining features. Execution time was about 4% faster using fixed variances. When the iterative algorithm was used, a stable solution was found after three iterations, giving the same result as in section 5.1.

## 6 Conclusions

The work of Christmas, Kittler and Petrou [2, 1] introduced probabilistic schemes for geometric feature matching in which the form of the compatibility coefficients was explicitly stated. However, they used a relatively simple noise model in order to evaluate them, which required the setting of several parameters; it was not clear what values these parameters should have. In this paper we provide an improved noise model for 2-D applications in which the image and model are related by an unknown Euclidean transformation. We find that the new noise model is an improvement over the simpler one in several respects:

- there are generally fewer unknown parameters (two in the example given);
- a rationale for calculating their values is provided, so that no training session is required, and
- there were fewer null labels when the non-iterative algorithm was used.

For a given number of iterations, execution time is very similar.

We have also suggested how other small deformations (*e.g.* scaling errors) can be incorporated, provided some statistical information for the deformations is available.

## References

- [1] W.J. Christmas, J. Kittler, and M. Petrou. Matching in computer vision using probabilistic relaxation. To appear in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1995.
- [2] W.J. Christmas, J. Kittler, and M. Petrou. Non-iterative contextual correspondence matching. In J-O. Eklundh, editor, *Computer Vision – ECCV'94*, volume II, pages 137–142, Stockholm, Sweden, 1994. Springer-Verlag.
- [3] R. Deriche, R. Vaillant, and O.D. Faugeras. From noisy edge points to 3D reconstruction of a scene - a robust approach and its uncertainty analysis. In P. Johansen and S. Olsen, editors, *Theory & applications of image analysis*, volume 2 of *Series in Machine Perception and Artificial Intelligence*, pages 71–78, Aalborg, Denmark D9108, 1992. World Scientific Publ. Co. PTE Ltd., Singapore.
- [4] R.A. Hummel and S.W. Zucker. On the foundations of relaxation labeling processes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 5(3):267–286, May 1983.
- [5] J. Kittler, W.J. Christmas, and M. Petrou. Probabilistic relaxation for matching problems in computer vision. In *Proceedings of the Fourth International Conference on Computer Vision*, pages 666–673, Berlin, Germany, 1993.
- [6] J. Kittler and E.R. Hancock. Combining evidence in probabilistic relaxation. *International Journal of Pattern Recognition and Artificial Intelligence*, 3:29–51, 1989.
- [7] M. Pelillo and M. Refice. Learning compatibility coefficients for relaxation labelling processes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 16:933–945, 1994.
- [8] A. Rosenfeld, R. Hummel, and S. Zucker. Scene labeling by relaxation operations. *IEEE Trans. Systems, Man, and Cybernetics*, 6:420–433, June 1976.
- [9] A.J. Stoddart, M. Petrou, and J. Kittler. Probabilistic relaxation as an optimizer. Accepted by BMVC'95, the Sixth British Machine Vision Conference, 1995.