

On using priors in affine matching

Venu Madhav Govindu^{a,*}, Michael Werman^b

^a*HIG-25, Simhapuri Layout, Visakhapatnam, AP 530047, India*

^b*Department of Computer Science Hebrew University of Jerusalem, Jerusalem, Israel*

Received 20 February 2003; received in revised form 25 February 2004; accepted 17 March 2004

Abstract

In this paper, we consider the generative model for affine transformations on point sets and show how a priori information on the noise and the transformation can be incorporated into the model resulting in more accurate algorithms. While invariants have been widely used, the existing literature fails to fully account for the uncertainties introduced by both noise and the transformation. We show how using such priors leads to algorithms for Bayesian estimation and a probabilistic interpretation of invariants which addresses the limitations of current methods. We present synthetic and real results for object recognition, image registration and determining object planarity to demonstrate the power of using priors for image comparison.

© 2004 Elsevier B.V. All rights reserved.

Keywords: Affine transformations; Affine invariants; Probabilistic models; Recognition

1. Introduction

In this paper we show how we can incorporate knowledge of both the transformation and noise priors into a probabilistic analysis of the affine point generative model. This model leads to different estimators, namely a Bayesian estimate of posterior probability and a probabilistic interpretation of the affine invariant. We show how using such priors improves the performance of the algorithms for registering, matching and comparing images. Two of the main criteria for comparing images or images to models are registration error and invariants. These methods have a long history [1,6] and together with image based representations make up a substantial part of image pattern recognition techniques. Algorithms that use registration find the transformation that minimises a given residual error. The differences between methods are the transformations (i.e. Projective, Affine, Euclidean etc.) and the error metrics used. In contrast, invariants are functions of points that are independent of the transformation and affine invariants are well studied as a tool for matching and indexing [9,6]. The affine model is also useful since planes under a weak perspective camera model behave in an affine manner.

The limitation of standard techniques is that they do not correctly account for data noise. Thus in the case of registration, the commonly used least squares metric might be inappropriate. Similarly for affine invariants, invariance does not hold when the data is noisy. In such a case, the estimate of the invariant will depend on both the amount of noise present and the applied affine transformation. Often, for object recognition the invariant is computed and matched with a set of models and the model, which is closest to the estimate in a Euclidean sense (i.e. using least squares of the difference) is declared the winner. This is ad-hoc and can only be justified by computational ease. There have been a number of papers under the name of shape space in the statistical literature (e.g. [5]) and in the computer vision literature that have studied the impact of noise on the invariant in order to improve recognition rates [4] and indexing [7]. However, these methods do not fully account for all available prior information. While [4] introduces a probabilistic affine invariant its analysis only considers the effect of noise on the invariant and does not incorporate information about the transformations. In Ref. [3], a related method is proposed for computing the Maximum Likelihood Estimate of the transformation. We point out that the effect of noise on the invariant will also depend on the scale of the transformation. If the transformation is large then the relative impact of the noise is small

* Corresponding author.

E-mail addresses: venu@narmada.org (V.M. Govindu), werman@cs.huji.ac.il (M. Werman).

and vice versa. Thus this relative effect of the transformation will have to be accounted for in a probabilistic setting by means of a prior on the affine transformations.

The rest of the paper is organised as follows. Section 2 briefly describes the generative model for our case and Section 3 describe the different estimators that follow from a probabilistic interpretation of the generative model. Section 4 describes the results of applying our methods to the problems of object recognition, image registration etc. and Section 5 will present some conclusions.

2. Generative model for affine points

In this section we describe the generative model for affine-transformed points. The observed two-dimensional points \mathbf{y} are generated by an affine transformation on a model \mathbf{m} and is corrupted by additive Gaussian noise. Hence, $\mathbf{y} = \mathbf{A}\mathbf{m} + \mathbf{n}$ where \mathbf{A} is the 2×2 affine transformation matrix¹ applied to the model \mathbf{m} and \mathbf{n} is the Gaussian noise added with $\mathbf{n} \sim N(\mathbf{0}, \Sigma_{\mathbf{n}})$. Similarly, the affine transformations are assumed to come from a Gaussian distribution i.e. $\mathbf{A} \sim N(\mu_{\mathbf{A}}, \Sigma_{\mathbf{A}})$. It must be kept in mind that the Gaussian assumption of the transformation model prior is only for analytic purposes and we can easily account for non-Gaussian priors by expressing this prior as a mixture of Gaussians. We also remark that the problem of learning meaningful priors is beyond the scope of this paper. In subsequent analysis, we will examine the effect of both the transformation and noise priors on the estimation process.

3. Estimation methods

In this section, we will describe different estimation methods as applied to our generative model.

3.1. Bayesian estimation method

Since the residual error is $\mathbf{d} = \mathbf{y} - \mathbf{A}\mathbf{m}$ and we have a Gaussian noise model, the conditional probability of the observed data given the model and the transformation is $P(\mathbf{y}|\mathbf{A}, \mathbf{m}) = e^{-1/2(\mathbf{y}-\mathbf{A}\mathbf{m})^T \Sigma_{\mathbf{n}}^{-1}(\mathbf{y}-\mathbf{A}\mathbf{m})}$. We can rewrite the term $\mathbf{A}\mathbf{m} = \mathbf{M}\mathbf{a}$, where \mathbf{a} is the column-ordered vector containing the terms in \mathbf{A} and \mathbf{M} is the appropriate matrix that contains elements of \mathbf{m} . Thus we can rewrite the conditional probability given above as

$$P(\mathbf{y}|\mathbf{A}, \mathbf{m}) = e^{-1/2(\mathbf{y}-\mathbf{M}\mathbf{a})^T \Sigma_{\mathbf{n}}^{-1}(\mathbf{y}-\mathbf{M}\mathbf{a})} \quad (1)$$

¹ While the affine transformation has six parameters, the translation terms do not affect the invariants. Hence to ensure a uniform comparison we remove the translation term from our model. It can be easily incorporated if required.

² There is a normalising term that will make this a true probability distribution. However, unless explicitly required in our analysis we will drop this normalising constant for notational convenience.

In our generative model the affine transformations are drawn from a Gaussian distribution, which implies that $\mathbf{a} \sim N(\mu_{\mathbf{a}}, \Sigma_{\mathbf{a}})$. Therefore, the posterior probability of observing the points given a model \mathbf{m} is obtained by *integrating out* the affine transformation by means of its prior, i.e.

$$\begin{aligned} P(\mathbf{y}|\mathbf{m}) &= \int P(\mathbf{y}|\mathbf{A}, \mathbf{m})P(\mathbf{A})d\mathbf{A} \\ &= \int e^{-1/2(\mathbf{y}-\mathbf{M}\mathbf{a})^T \Sigma_{\mathbf{n}}^{-1}(\mathbf{y}-\mathbf{M}\mathbf{a})} e^{-1/2(\mathbf{a}-\mu)^T \Sigma_{\mathbf{a}}^{-1}(\mathbf{a}-\mu)} d\mathbf{a} \end{aligned} \quad (2)$$

The exponent in Eq. (2) is quadratic in the affine transformation \mathbf{a} and hence can be solved easily by completion of squares. For the problem of object recognition if we have two models \mathbf{m}_1 and \mathbf{m}_2 , we can compute the conditional probabilities, $P(\mathbf{y}|\mathbf{m}_1)$ and $P(\mathbf{y}|\mathbf{m}_2)$ and classify according to whichever likelihood value is higher. In Ref. [2], a similar prior is used to control the estimate of an affine transformation between two point sets.

3.2. Affine invariants

To compute affine invariants we use the first three model points as the basis (i.e. $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$). Therefore any point \mathbf{m} is described by its co-ordinates (α, β) in the invariant space. These co-ordinates satisfy the relationship,

$$\mathbf{m} - \mathbf{m}_1 = \alpha(\mathbf{m}_2 - \mathbf{m}_1) + \beta(\mathbf{m}_3 - \mathbf{m}_1) \quad (3)$$

The relationship in Eq. (3) can be seen to be invariant to the application of an affine transformation on the model points since $\mathbf{m} - \mathbf{m}_1 = \alpha(\mathbf{m}_2 - \mathbf{m}_1) + \beta(\mathbf{m}_3 - \mathbf{m}_1) \Rightarrow \mathbf{A}\mathbf{m} - \mathbf{A}\mathbf{m}_1 = \alpha(\mathbf{A}\mathbf{m}_2 - \mathbf{A}\mathbf{m}_1) + \beta(\mathbf{A}\mathbf{m}_3 - \mathbf{A}\mathbf{m}_1)$. The ‘naive’ way of using the affine invariants for object recognition is to compute the affine invariants ($\mathbf{c} = (\alpha, \beta)$) for a given set of observed feature points \mathbf{y} and compare them with the model co-ordinates \mathbf{c}_1 and \mathbf{c}_2 . The model closest to \mathbf{c} is chosen as the classification. As we shall show in the next subsection this method fails to satisfactorily account for the effect of the noise and the transformation on the estimated invariant (in particular one must note the effect on the basis points).

3.3. Probabilistic interpretation of invariant

In our formulation, the k th feature point is given by $\mathbf{y}_k = \mathbf{M}_k \mathbf{a} + \mathbf{n}_k$ and by definition of the invariant, we have $\mathbf{y}_k = (1 - \alpha_k - \beta_k)\mathbf{y}_1 + \alpha_k \mathbf{y}_2 + \beta_k \mathbf{y}_3$. Consequently, the noise term in the k th point can be expressed as

$$\begin{aligned} \mathbf{n}_k &= \mathbf{y}_k - \mathbf{M}_k \mathbf{a} \\ &= (1 - \alpha_k - \beta_k)\mathbf{y}_1 + \alpha_k \mathbf{y}_2 + \beta_k \mathbf{y}_3 - \mathbf{M}_k \mathbf{a} \\ &= [(1 - \alpha_k - \beta_k)\mathbf{M}_1 + \alpha_k \mathbf{M}_2 + \beta_k \mathbf{M}_3 - \mathbf{M}_k] \mathbf{a} \\ &\quad + [(1 - \alpha_k - \beta_k)\mathbf{n}_1 + \alpha_k \mathbf{n}_2 + \beta_k \mathbf{n}_3] \end{aligned} \quad (4)$$

This implies that given the object model and the affine co-ordinates, the ‘estimated’ noise in any feature point depends on the four parameters of the affine transformation (\mathbf{a}) and the six parameters of the noise in the basis points (i.e. in $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$). Therefore, we have the following conditional probability for $P(\mathbf{n}_k|\mathbf{M})$,

$$\int e^{-1/2\mathbf{n}_k^T \sum_n^{-1} \mathbf{n}_k} P(\mathbf{a})P(\mathbf{n}_1)P(\mathbf{n}_2)P(\mathbf{n}_3) d\mathbf{a}d\mathbf{n}_1 d\mathbf{n}_2 d\mathbf{n}_3 \quad (5)$$

where the term \mathbf{n}_k is as given in Eq. (4). However, the probability that we are interested in is $P(\alpha_k, \beta_k|\mathbf{M})$. Thus we transform the probability distribution from \mathbf{n}_k to that of (α_k, β_k) by a change of variables. This uses the Jacobian of the transformation between the two variables, i.e. $|\mathbf{J}|$ between \mathbf{n}_k and $(\alpha_k, \beta_k|\mathbf{M})$. Now to express the required probability as an integral, we concatenate the affine transformation and the noise terms into a single vector, $\mathbf{x} = [\mathbf{a}, \mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3]$. Therefore,

$$P(\alpha_k, \beta_k|\mathbf{M}) = e^{-1/2s} \int e^{-1/2(\mathbf{x}-\mu_x)^T \sum_x^{-1} (\mathbf{x}-\mu_x)} |\mathbf{J}| d\mathbf{x} \quad (6)$$

where \mathbf{J} is the required Jacobian matrix and s is a constant term. From Eq. (4) we see that

$$\frac{\partial \mathbf{n}_k}{\partial \alpha_k} = [\mathbf{M}_2 - \mathbf{M}_1] \mathbf{a} + \mathbf{n}_2 - \mathbf{n}_1 = \mathbf{L}_\alpha \mathbf{x}$$

$$\frac{\partial \mathbf{n}_k}{\partial \beta_k} = [\mathbf{M}_3 - \mathbf{M}_1] \mathbf{a} + \mathbf{n}_3 - \mathbf{n}_1 = \mathbf{L}_\beta \mathbf{x}$$

where \mathbf{L}_α and \mathbf{L}_β are appropriate matrices. Since the above partial derivatives can be expressed as linear constraints in \mathbf{x} , the entire Jacobian can be represented as a quadratic

expression in \mathbf{x} , i.e. $|\mathbf{J}| = |\mathbf{x}^T \mathbf{B} \mathbf{x}|$. But we have $\mathbf{N} - 3$ affine co-ordinates that are being transformed, making the effective transformation $|\mathbf{J}|^{\mathbf{N}-3}$. Therefore, the resultant form for the probability function $P(\alpha, \beta|\mathbf{M})$ is

$$e^{-1/2s} \int e^{-1/2(\mathbf{x}-\mu_x)^T \sum_x^{-1} (\mathbf{x}-\mu_x)} |\mathbf{x}^T \mathbf{B} \mathbf{x}|^{\mathbf{N}-3} d\mathbf{x} \quad (7)$$

where (α, β) represents the affine co-ordinates for the observed points. This formulation is similar to that of Ref. [4]. However, the affine transformation prior is also included in our analysis.

In our solution to Eq. (7) adopted from Ref. [4], the absolute value is dropped thereby providing an approximation when \mathbf{N} is even since then $\mathbf{N} - 3$ is odd. This approximation is reasonable only when the covariances \sum_a and \sum_n are small. However for odd powers of \mathbf{N} this solution is exact. The reader is referred to [4] for details.

Also we would like to address the issue of non-Gaussian priors for the affine transformation, a situation that arises in real life. Often we can reasonably approximate $P(\mathbf{A})$ as a mixture of Gaussians, i.e. $P(\mathbf{A}) = \sum_i m_i N(\mu_i, \Sigma_i)$ where m_i is the relative mixing proportion and $N(\mu, \Sigma)$ denotes a Gaussian distribution. As can be easily seen from Eqs. (2) and (5), we can incorporate this non-Gaussian prior into the analysis due to the linearity of the integral operator.

To demonstrate the correctness of the probabilistic invariant (Eq. (7)) we compare its distribution with that of an empirically derived one in Fig. 1. In this comparison, we use a 4 point model which has an affine shape of (1,2), i.e. $\mathbf{m} = [(0, 0), (1, 0), (0, 1), (1, 2)]$. The mean of the affine

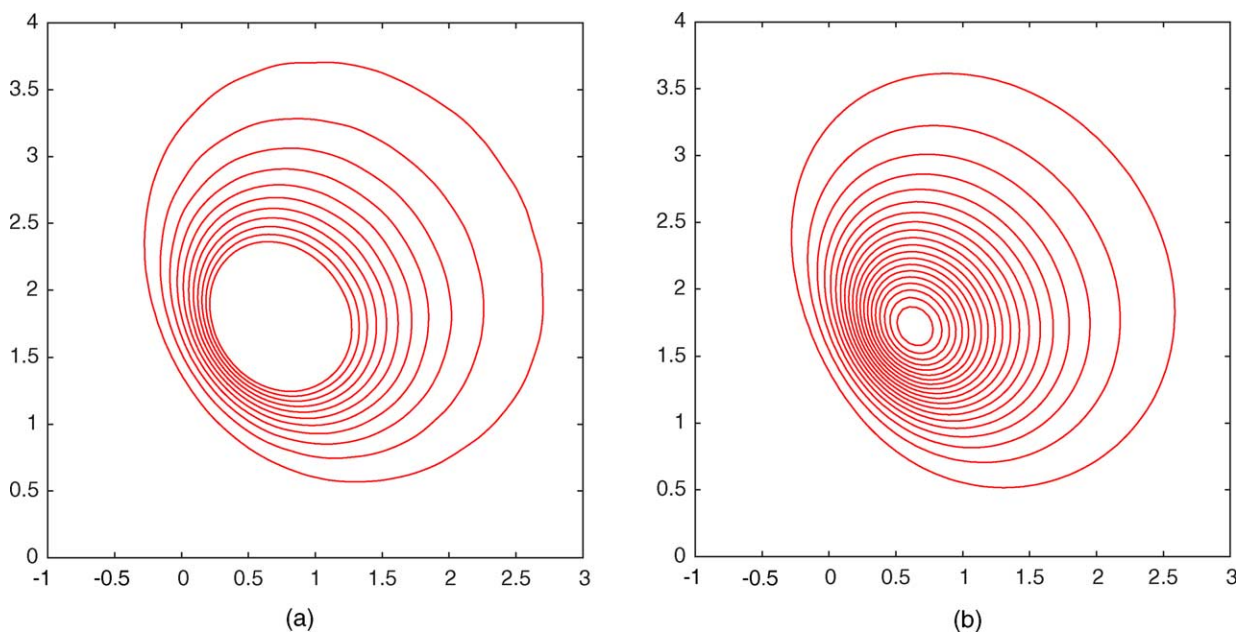


Fig. 1. (a) shows the empirically derived affine shape distribution for a given model. The analytic expression for the probability distribution is shown in (b) and is seen to be identical to the empirical distribution. The cross-correlation between the two distributions is 0.999 showing that our derivation of the analytic distribution is correct.

transformation prior is

$$\mu_{\mathbf{a}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

and its covariance is $\Sigma_{\mathbf{a}} = 0.02\mathbf{I}_4$, where \mathbf{I}_4 is a 4×4 identity matrix. However the additive noise used is non-white and has a covariance of

$$\Sigma_{\mathbf{n}} = 0.05 \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix}.$$

For the empirically derived distribution, we use the parameters given above to compute the affine shape 5 million times and the resultant probability distribution of the affine shape is shown in Fig. 1(a). This can be compared with the analytically derived distribution for the same case which is shown in Fig. 1(b). As can be seen the two probability distributions are identical and in fact the cross-correlation coefficient of the two distributions is 0.999. This clearly demonstrates that the analytic expression for the probabilistic invariant is correct inspite of the approximation made due to the dropping of the sign of the Jacobian in our analysis.

Finally, in the standard least squares method, the model with the smallest residual error, ($\mathbf{d} = \|\mathbf{y} - \mathbf{A}\mathbf{m}\|^2$) is selected. Here \mathbf{A} is the linear estimate of \mathbf{A} .

4. Experiments

In this section we will describe experiments with synthetic and real data that demonstrate the power of explicitly incorporating priors into the generative model for object recognition and image comparison.

4.1. Recognition accuracy

In this subsection, we will describe the performance of the different algorithms for object recognition. We will briefly describe the experimental protocol used and show the results than can elucidate the behaviour of the different recognition methods.

For our experiments we used point sets that range from 4 to 10 points in each data set. For each case we generated two models and performed recognition using the various algorithms. Our experiments are symmetric, i.e. for each pair of models generated, we test for recognition accuracy with one instance of each model generating a data set. The error rates are averaged over 1000 trials (i.e. the averaging is over $(10 - 3) \times 1000 \times 2 = 14,000$ experiments).

In our experiments, not only do we look at the performance of the different algorithms but we are also interested in looking at the effect of incorporating the priors into our models. This is of importance since we want to

demonstrate the power of using such priors in recognition and comparison. The models \mathbf{m}_1 and \mathbf{m}_2 are generated by picking $N - 3$ affine co-ordinates (the other three points being the canonical basis) using a Gaussian distribution with a mean 0 and variance of 5. Now for each instance, we do not simply pick an affine transformation \mathbf{A} and noise \mathbf{n} from fixed distributions. Instead we first pick priors for the transformation and noise and then use them to randomly pick instances of the transformation and noise. The ranges for the transformation prior and noise are $[0,5]$ and $[0,0.5]$ respectively. Therefore for each instance, we first pick the quantities $\Sigma_{\mathbf{A}}$ and $\Sigma_{\mathbf{n}}$ uniformly from these ranges. Thus we now construct two priors $\Sigma_{\mathbf{A}} = \sigma_{\mathbf{A}}^2 \mathbf{I}_{4 \times 4}$ and $\Sigma_{\mathbf{n}} = \sigma_{\mathbf{n}}^2 \mathbf{I}_{n \times n}$ where $\mathbf{I}_{n \times n}$ is an n -dimensional identity matrix. Thereafter we draw an affine transformation \mathbf{A} and noise values \mathbf{n} from $\Sigma_{\mathbf{A}}$ and $\Sigma_{\mathbf{n}}$, respectively and generate data points $\mathbf{y} = \mathbf{A}\mathbf{m}_i + \mathbf{n}$ where $i \in \{1, 2\}$, i.e. each of the two models are used once.

Since the error rate for the Bayesian method is always the lowest, we use this as a lower bound and show the relative errors by dividing each of the error rates by the Bayesian error rate. This allows us to focus on the relative performance of each method without having to account for the actual error rates which will vary according to the dimensionality of the problem (i.e. vary with the number of points involved). In Fig. 2(a) we show the relative error rates for the different methods that are appropriately labelled. The method due to Leung et al. [4] is also shown for comparison. Obviously the relative Bayesian error rate is always 1.

The plot labelled ‘naive invariant’ is one where the invariant for the data set is computed and compared with the two models to find the closest one in the Euclidean sense. This is of course the standard method of using an invariant for recognition without using any prior information and expectedly does the worst amongst the different methods (as indicated by its high value of relative error). It can also be clearly seen that our probabilistic invariant (‘prob invariant’) does significantly better than ‘Leung’s method’ due to the fact that our generative model and the subsequent analysis in Section 3.3 explicitly incorporates priors for both the affine transformation \mathbf{A} and the noise \mathbf{n} .

It bears repeating that just the way we use a Gaussian prior for data noise (to reflect the fact that large noise values are less likely) the knowledge that certain affine transformations are less likely than others will have to be explicitly accounted for in our model. This is obviously important since in the process of computing the invariant the data is scaled by an estimated affine transformation implying that the scale of the affine transformation will determine the impact of noise on the accuracy of the invariant computed. Thus in a truly probabilistic analysis, we will need to account for the transformation prior as is the case with our probabilistic invariant. In contrast, Leung’s method cannot use the prior information of the affine transformation and is limited to using the knowledge of the noise prior. It is also

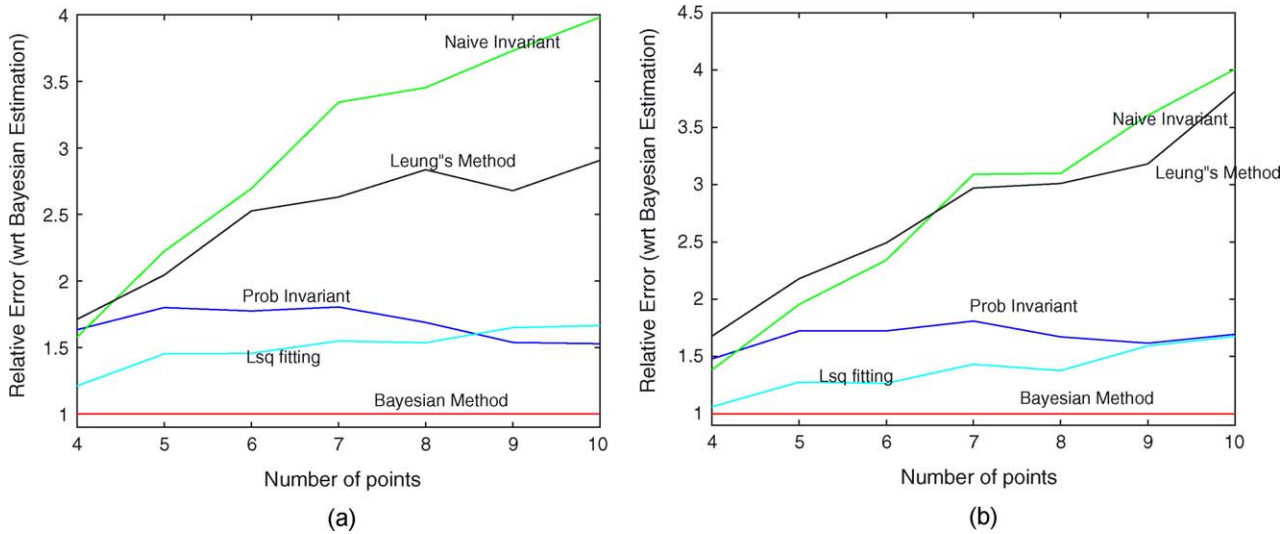


Fig. 2. The relative error rates of each recognition algorithm are shown. In (a) we use the full information of the underlying priors; (b) uses a fixed prior.

worth noting that a simple least squares estimation method (that does not use any priors) does better or as well as the probabilistic invariant method. This could probably be attributed to the loss of information that results when we compress the N point data into $N - 3$ affine co-ordinates (i.e. the invariant). There is no such compression of information in the full Bayesian method resulting in the highest accuracy. However, in the event we are interested in computing an invariant and using it for object recognition, our experiments demonstrate that we should use all the prior information available and incorporate it into our probabilistic analysis.

In the results discussed above (Fig. 2(a)) for each instance of recognition, we used the correct priors (i.e. we provided the Bayesian method, Leung's method and our probabilistic invariant method with the covariances Σ_A and Σ_n). Note that we only provide them with the correct prior and not the actual values of transformation and noise that are randomly drawn from these priors. However in a real-life situation we will have to take recourse to using a fixed prior. Thus for the results shown in Fig. 2(b) we show the recognition results for the same data set but using fixed priors that are the average of the different priors used. We may remind the reader that both the 'naive invariant' and 'lsq fitting' methods do not use the prior information and hence their error rates do not change. However it is interesting to note that Leung's method's performance becomes similar to that of the naive invariant while our probabilistic method does not change that much in performance. Thus while expectedly both methods will do worse here (since we have less knowledge of the underlying priors), our method performs better than that of Ref. [4].

4.2. Likelihood ratios for matching sets

To use the probabilities defined earlier for hypothesis testing we will have to compare them with a threshold.

Hence we need to 'normalise' the probabilities for meaningful thresholds to be defined. In the case of the Bayesian method of Eq. (2) we have a conditional probability which can be extended to a likelihood. Thus, given two point sets y_1 and y_2 we can define their Bayes likelihood as $L_b(y_1, y_2)P(y_1|y_2)/P(y_2|y_1)$. However a symmetric Bayesian Likelihood can be defined as

$$L_b(y_1, y_2) \sqrt{\frac{P(y_1|y_2)P(y_2|y_1)}{P(y_1|y_1)P(y_2|y_2)}} \quad (8)$$

This ratio is symmetric ($L_b(y_1, y_2)L_b(y_2, y_1)$) and is normalised to lie in the range $[0, 1]$. The likelihood for the probabilistic invariant can be similarly defined. These likelihood ratios can be used to measure the confidence we have that two given point sets arise from the same underlying model. It must be emphasised that this likelihood measure does not depend on knowing the underlying model at all, rather it simply defines a probability-like measure that two observed data sets are from the same generative model. A high likelihood value implies a high 'match' confidence which lends itself to the following method for finding correspondences.

4.3. Correspondences in multi-sensor images

One common method for image registration is to match feature points and compute the relative transformation between the two images [1]. In general, computing feature correspondences is a hard task and is further compounded for images from different sensors as there is no obvious radiometric relationship between the images (see Fig. 3). Here we have to rely on the geometry of the images to establish correspondences.

In this example, we demonstrate the use of the likelihoods to establish correspondences between feature points in the two images. Using a simple corner detector we

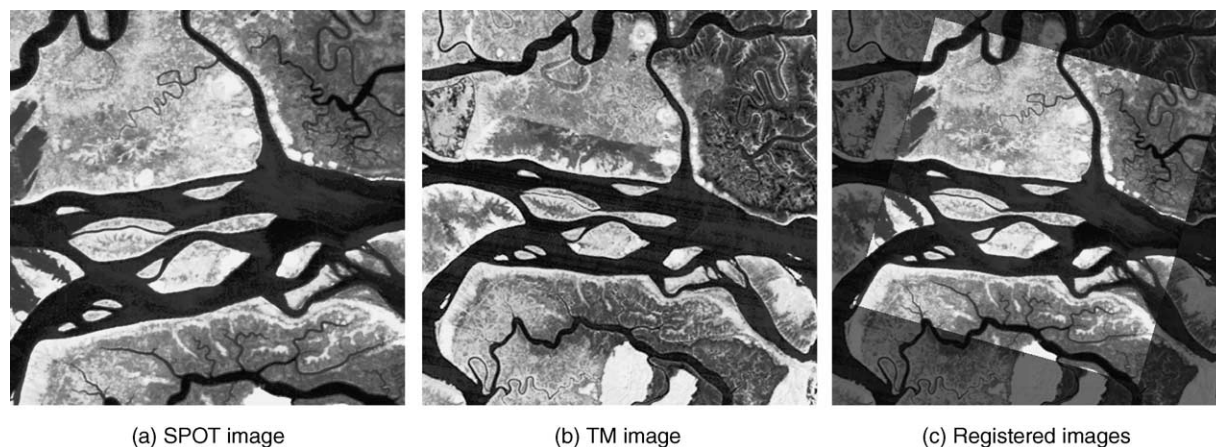


Fig. 3. Registration of multi-sensor images using likelihood to derive correspondences.

extract 300 ‘interest points’ from each image. We manually select 3 correspondences for a basis set for the affine invariants. Subsequently we automate the process of deriving more feature correspondences. For points $\{X_1\}$ and $\{X_2\}$ in the two images, every tuple in the set $\{X_1 \times X_2\}$ is a potential correspondence but this set can be pruned using the bases to limit the search space (say within d pixels after transformation). For every x_1 in the first image, we compute the likelihoods of its possible matches in the second set and select the one with the highest likelihood value and finally disambiguate multiple matches in the correspondence set. This is significantly faster since now our search complexity is $O(N)$ instead of $O(N^2)$ for N interest points in each image. The results of the registration obtained using 46 ‘discovered’ correspondences are shown in Fig. 3(c) and can be seen to be very accurate. The root mean square registration error is 0.95 pixels,³ As a control test, we used the basis points to warp one point set onto the other and picked the closest match to select correspondences which resulted in 46 correspondences with a slightly higher error of 0.96 pixels. This error is higher since some of the correspondences obtained here were wrong. In contrast, our model more accurately captures the notion of likelihood of point matches. It is significant that our process is automatic since obtaining feature correspondences in a multi-sensor scenario (esp. with large scale changes) is difficult.

4.4. Measuring coplanarity

While in Section 4.1, we considered recognition accuracy, here we focus on using the likelihood measures for another task, i.e. verifying if a point set is affine-transformed. When the points lie on a plane and the camera is roughly weak-perspective, we expect the points to behave in an ‘affine’ manner, i.e. their relative transformations will be sufficiently captured by an affine transformation.

Thus the goodness of affine fit of the data is a measure of how close the data is to being planar and can be used to guide image segmentation. In Ref. [8], planar invariants are used (albeit in a non-probabilistic sense) to group coplanar points for ground plane detection.

We will illustrate our results using two sequences from the familiar COIL database from Columbia University, (Figs. 4 and 5) which we call ‘Anacin’ and ‘Piggybank’ respectively. The Anacin images consist of planes and the Piggybank is a non-planar surface. In both these examples, the objects were placed on a turntable and rotated by one complete revolution in fixed steps. For our purposes we use 13 images from each sequence since the areas being viewed disappear beyond the range of these images. We use a conventional image-matching scheme to match and track feature points over the entire sequence.

For the Anacin sequence, we use a conventional image-matching scheme to match feature points across the entire sequence. This results in 37 feature points being tracked over the entire sequence. The feature points tracked are marked on Fig. 4(a). The points used as the basis are shown by the co-ordinate frame on the vertical plane.

In Fig. 4(b) we show the Bayesian likelihood ratio of these 37 points (i.e. 34 points using the rest of the 3 points as basis points) for the entire sequence (This is shown in blue as a continuous plot and is indicated by the legend ‘two planes’). In the same figure, we also show the likelihood ratio when we consider only those points that lie on the vertical plane of the Anacin box (in red and marked with diamonds with the legend ‘single plane’). The likelihood ratio shown in the experiments of this subsection are the Bayesian likelihood of each point set in each image as compared to the same feature points in the first image. In other words, if we denote the tracked feature points in image k as \mathbf{p}_k , we are measuring the likelihood ratio, $L_b(\mathbf{p}_k, \mathbf{p}_1)$. Thus obviously, the likelihood values for the first image are equal to 1, since the point set is the same as the point set used as the model.

As may be observed, both the likelihood values stay close to 1 for most of the sequence and taper off towards

³ The results for both definitions of likelihood are identical in this case. Also, we use the same data set to derive the transformation and the noise priors. In a case with many image sets the underlying priors can be learnt.

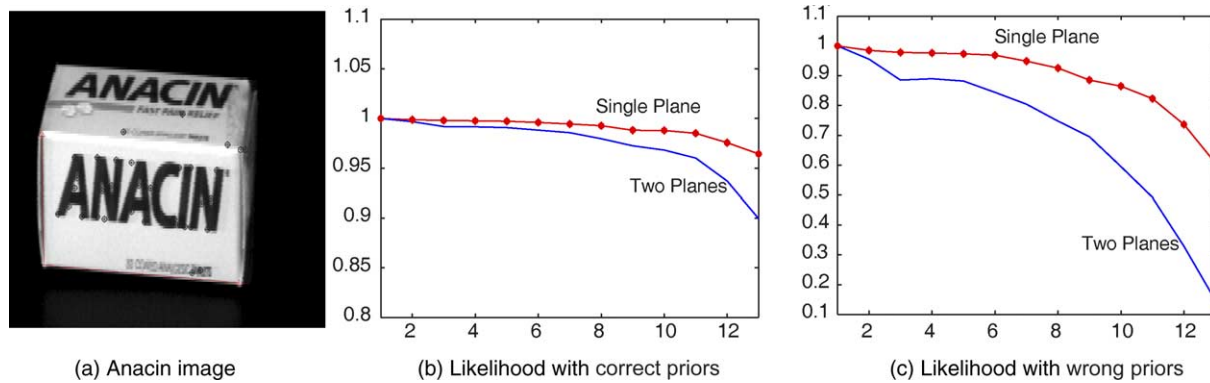


Fig. 4. Likelihoods for 'Anacin' sequence.

the end of the sequence. This tapering off is due to the fact that towards the end of the sequence, the vertical plane is almost parallel to the z -axis of the camera (i.e. viewing direction) and hence the perspective effects do get pronounced. In other words, for most of the sequence all the feature points lie roughly at the same depth and only start varying in depth towards the end of the sequence. The relative behaviour of the two plots is also interesting. In the case where all the feature points are confined to the same plane (as the basis points) we get a better likelihood ratio than when some of the points happen to lie on a different plane. This behaviour is to be expected since the instance of 'single plane' better confirms to an affine transformation on the data points and hence the likelihood ratio is less varying than when the points confirm less to the coplanarity assumption.

In both these plots, we use the tracked feature points to estimate the correct priors for the affine transformation and the noise in the data. The likelihood measures shown are using these priors. We also illustrate the effect of using a wrong prior set on the likelihood ratios in Fig. 4(c). As can be observed, the effect of a wrong prior is significant since the likelihood ratio falls off dramatically. However, as we have indicated, the correct priors for the sequence can be

easily inferred from the data set itself and does not need any external information.

In Fig. 5(b), we show the likelihood ratio of the Piggybank sequence using its own correct priors (shown in black dashed line). For the sake of comparison, we have also included the likelihood plots for the Anacin sequence from Fig. 4(b) in this plot. As can be easily observed, since the points on the Piggybank are not coplanar, the effect of the rotation of the object is pronounced. As the object rotates, the transformation between the tracked points and the points in the first image are less and less 'affine' like, since the effect of the non-planarity gets more and more pronounced. Thus the likelihood ratio falls off significantly.

In summary, a glance at Fig. 5(b) should tell us that the Piggybank is not a coplanar object while the points in one set of the Anacin sequence are coplanar. It also tells us that the second set of points in the Anacin sequence deviate slightly from coplanarity but significantly less than that of the Piggybank. Thus, our measure accurately captures the notion of co-planarity or 'affineness' for objects being considered. It may also be pointed out that while we have shown the Bayesian likelihood here, the likelihood ratio for the probabilistic invariant has similar behaviour.

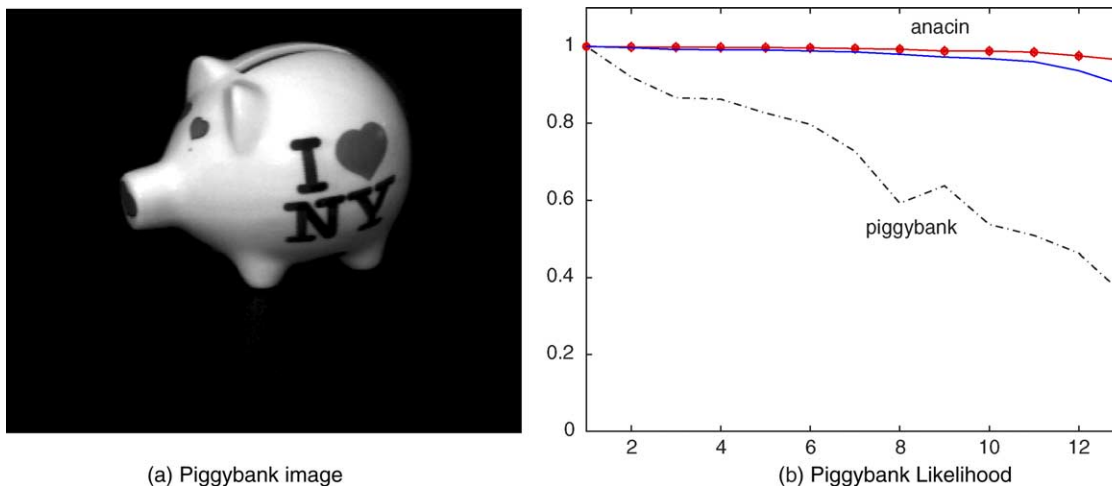


Fig. 5. Likelihoods for 'Piggybank' sequence.

5. Conclusions

In this paper, we have considered the generative model for affine transformations on image points. We have describe how the incorporation of appropriate priors of the transformation and noise into the generative model leads to better estimators. The use of these estimators are demonstrated on the problems of object recognition, image registration and comparison. It is observed that the Bayesian method outperforms all other methods and our formulation of the probabilistic invariant is preferable over others.

References

- [1] L.G. Brown, A survey of image registration techniques, *ACM Surveys* 24 (4) (1992) 325–376.
- [2] A. Fitzgibbon, A. Zisserman, On Affine Invariant Clustering and Automatic Cast Listing in Movies, *European Conference on Computer Vision III*, 2002, 304 pp..
- [3] D. Keren, I. Shimshoni, L. Goshen, M. Werman, All Points Considered: A Maximum Likelihood Method for Motion Recovery, *Theoretical Foundations of Computer Vision*, Springer LNCS Series, vol. 2616, 2003, pp. 72–85.
- [4] T. Leung, M. Burl, P. Perona, Probabilistic Affine Invariants for Recognition, *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 1998, pp. 678–684.
- [5] K.V. Mardia, I.L. Dryden, *Statistical Shape Analysis*, Wiley, London, 1998.
- [6] J.L. Mundy, A. Zisserman, *Geometric Invariance in Computer Vision*, MIT Press, Cambridge, MA, 1992.
- [7] I. Rigoutsos, R. Hummel, A Bayesian approach to model matching with geometric hashing, *Comput. Vision Image Understanding: CVIU* 62 (1) (1995) 11–26.
- [8] D. Sinclair, A. Blake, Quantitative planar region detection, *Int. J. Comput. Vision* 18 (1) (1996) 77–91.
- [9] H.J. Wolfson, Y. Lamdan, Geometric hashing: a general and efficient model-based recognition scheme, *International conference on computer vision*, 1988, pp. 238–249.