On Using Priors in Affine Matching

Venu Govindu

HIG-25, Simhapuri Layout
Visakhapatnam, INDIA
venu@cfar.umd.edu

Michael Werman
Department of Computer Science
Hebrew University of Jerusalem
Jerusalem, ISRAEL
werman@cs.huji.ac.il

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 the 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.*

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, 5]) and together with image based representations make up the bulk 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 (ie. 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 [8, 5]. 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 (ie. 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 (eg. [4] in the statistical literature) and in the computer vision literature that have studied the impact of noise on the invariant in order to improve recognition rates [3] and indexing [6]. However these methods do not fully account for all available prior information. While [3] introduces a probabilistic affine invariant its analysis only considers the effect of noise on the invariant and does not incorporate information about the transformations. 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 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. Sec. 2 briefly describes the generative model for our case and Sec. 3 describe the different estimators that follow from a probabilistic interpretation of the generative model. Sec. 4 describes the results of applying our methods to the problems of object recognition, image registration etc. and Sec. 5 will end with 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, $\mathbf{y} = \mathbf{Am} + \mathbf{n}$ where $\mathbf{A}$ is the 2×2 affine transformation matrix [1] 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. 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 $d = \mathbf{y} - \mathbf{Am}$ and we have a Gaussian noise model, the conditional probability of the observed data given the model and the transformation is $P(\mathbf{y}|\mathbf{A}, \mathbf{m}) = e^{-\frac{1}{2}(\mathbf{y}-\mathbf{Am})^{\mathbf{T}}\mathbf{\Sigma}_n{}^{-1}(\mathbf{y}-\mathbf{Am})}$ [2]. We can rewrite the term $\mathbf{Am} = \mathbf{Ma}$, 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

$$P(\mathbf{y}|\mathbf{A}, \mathbf{m}) = e^{-\frac{1}{2}(\mathbf{y}-\mathbf{Ma})^{\mathbf{T}}\mathbf{\Sigma}_n{}^{-1}(\mathbf{y}-\mathbf{Ma})} \tag{1}$$

Now, in our generative model the affine transformations are drawn from a Gaussian distribution, which implies that $\mathbf{a} \sim N(\mu_{\mathbf{a}}, \mathbf{\Sigma}_{\mathbf{a}})$.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.

$$\begin{aligned} P(\mathbf{y}|\mathbf{m}) &= \int P(\mathbf{y}|\mathbf{A}, \mathbf{m}) P(\mathbf{A}) d\mathbf{A} \qquad (2) \\ &= \int e^{-\frac{1}{2}(\mathbf{y}-\mathbf{Ma})^{\mathbf{T}}\mathbf{\Sigma}_n{}^{-1}(\mathbf{y}-\mathbf{Ma})} e^{-\frac{1}{2}(\mathbf{a}-\mu)^{\mathbf{T}}\mathbf{\Sigma}_a{}^{-1}(\mathbf{a}-\mu)} d\mathbf{a} \end{aligned}$$

The exponent in Eqn. 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 [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 (ie, $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$). Therefore any point $\mathbf{m}$ is described by its co-ordinates (α, β) in the invariant space. These co-ordinates satisfy the relationship,

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

The relationship in Eqn. 3 can be seen to be invariant to the application of an affine transformation on the model points since $\mathbf{m} - \mathbf{m}_1 = \alpha(\mathbf{m}_2 - \mathbf{m}_1) + \beta(\mathbf{m}_3 - \mathbf{m}_1) \Rightarrow \mathbf{Am} - \mathbf{Am}_1 = \alpha(\mathbf{Am}_2 - \mathbf{Am}_1) + \beta(\mathbf{Am}_3 - \mathbf{Am}_1)$. 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

Since in our formulation, the kth 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 kth point can be expressed as

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

This implies that given the object model and the affine co-ordinates, the “estimated” noise in any feature point depends on the 4 parameters of the affine transformation ($\mathbf{a}$) and the 6 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 $P(\mathbf{n}_k|\mathbf{M})$,

$$\int e^{-\frac{1}{2}\mathbf{n}_k{}^{\mathbf{T}}\mathbf{\Sigma}_{\mathbf{n}}{}^{-1}\mathbf{n}_k} P(\mathbf{a})P(\mathbf{n}_1)P(\mathbf{n}_2)P(\mathbf{n}_3)d\mathbf{a}d\mathbf{n}_1d\mathbf{n}_2d\mathbf{n}_3 \tag{5}$$

where the term $\mathbf{n}_k$ is as given in Eqn. 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

[1] While the affine transformation has 6 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

of (α_k, β_k) by means of 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,

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

where $\mathbf{J}$ is the required Jacobian matrix and $\mathbf{s}$ is a constant term. From Eqn. 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 $\mathbf{L}_\alpha$ and $\mathbf{L}_\beta$ 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. $|\mathbf{J}| = |\mathbf{x}^{\mathbf{T}}\mathbf{B}\mathbf{x}|$. 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

$$e^{-\frac{1}{2}\mathbf{s}} \int e^{-\frac{1}{2}(\mathbf{x}-\mathbf{mu_x})^{\mathbf{T}}\mathbf{\Sigma_x}^{-1}(\mathbf{x}-\mathbf{mu_x})}|\mathbf{x}^{\mathbf{T}}\mathbf{B}\mathbf{x}|^{\mathbf{N}-3}d\mathbf{x} \tag{7}$$

where (α, β) represents the affine co-ordinates for the observed points. This formulation is similar to that of [3]. However, the affine transformation prior is also included in our analysis.

In our solution to Eqn. 7 adopted from [3], 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 $\mathbf{\Sigma}_a$ and $\mathbf{\Sigma}_n$ are small. However for odd powers of $\mathbf{N}$ this solution is exact. The reader is referred to [3] 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. $P(\mathbf{A}) = \sum_i m_i G(\mu_i, \Sigma_i)$ where m_i is the relative mixing proportion and $G(\mu, \Sigma)$ denotes a Gaussian. As can be easily seen from Eqns. 2 and 5, we can incorporate this non-Gaussian prior into the analysis due to the linearity of the integral operator. We are unable to present results on this aspect of the problem in this paper due to space constraints.

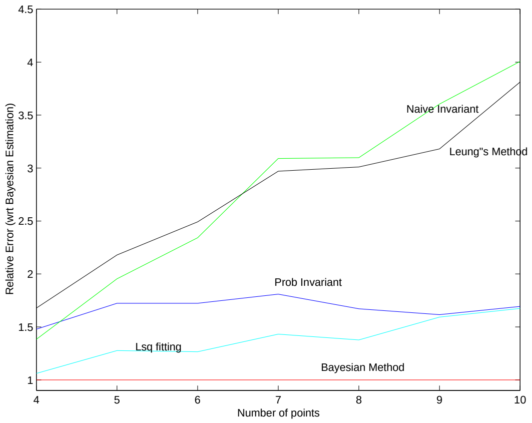


Figure 1: The relative error rates of each method.

Finally, in the standard least squares method, the model with the smallest residual error, $(\mathbf{d} = ||\mathbf{y} - \hat{\mathbf{A}}\mathbf{m}||^2)$ is selected. Here $\hat{\mathbf{A}}$ 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. These 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 3 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 $\sum_{\mathbf{A}} = {\sigma_{\mathbf{A}}}^2 I_{4\times 4}$ and $\sum_{\mathbf{n}} = {\sigma_{\mathbf{n}}}^2 I_{n\times n}$ where $I_{n\times n}$ is an n-dimensional identity matrix. Thereafter we draw an affine transformation $\mathbf{A}$ and noise values $\mathbf{n}$ from $\sum_{\mathbf{A}}$ and $\sum_{\mathbf{n}}$ 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. with the number of points). In Fig. 1 we show the relative error rates for the different methods that are appropriately labeled. The method due to Leung et al [3] is also shown for comparison. Obviously the relative Bayesian error rate is always 1. The plot labeled "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 Sec. 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 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 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 (ie. 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.

The results in Fig. 1 show the error rates that uses a fixed prior which is the average of the priors used, since in real-life we do not always have full knowledge of the underlying prior. It is interesting to note that Leung's method's performance is similar to that of the naive invariant while our probabilistic method has better performance.

4.2. Likelihood Ratios for matching sets

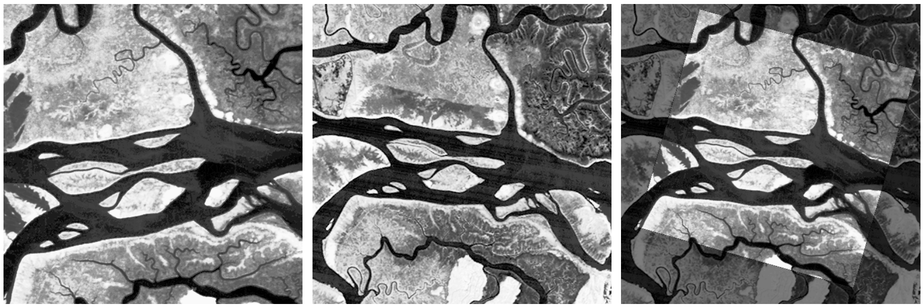
To use the probabilities defined earlier for hypothesis testing we will have to compare them with a threshold. Hence we need to "normalise" the probabilities for meaningful thresholds to be defined. In the case of the Bayes method of Eqn. 2 we have a conditional probability which can be extended to a likelihood. Thus, given two point sets $\mathbf{y}_1$ and $\mathbf{y}_2$ we can define their Bayes likelihood as $L_b(\mathbf{y}_1, \mathbf{y}_2) = \frac{P(\mathbf{y}_1|\mathbf{y}_2)}{P(\mathbf{y}_2|\mathbf{y}_2)}$. However a symmetric Bayesian Likelihood can be defined as

$$L_b(\mathbf{y}_1, \mathbf{y}_2) = \sqrt{\frac{P(\mathbf{y}_1|\mathbf{y}_2)P(\mathbf{y}_2|\mathbf{y}_1)}{P(\mathbf{y}_1|\mathbf{y}_1)P(\mathbf{y}_2|\mathbf{y}_2)}} \qquad (8)$$

This ratio is symmetric ($L_b(\mathbf{y}_1, \mathbf{y}_2) = L_b(\mathbf{y}_2, \mathbf{y}_1)$) 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 features 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



(a) SPOT image (b) TM image (c) Registered Images

Figure 2: Registration of multi-sensor images using likelihood to derive correspondences.

radiometric relationship between the images (See Fig. 2). Here we have to rely on the geometry of the images to establish correspondences.

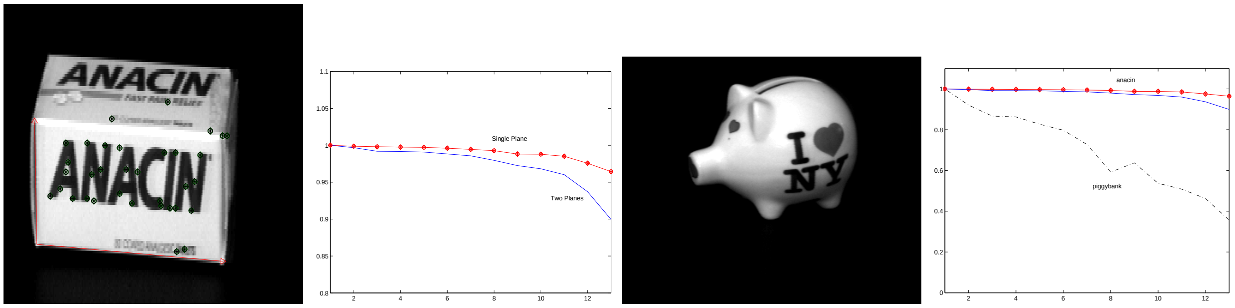
In this example, we demonstrate the use of the likelihoods to establish correspondences between feature points in the two images. Using a simple corner detector we extract 300 "interest points" from each image. We manually select 3 correspondences for a basis set for the affine invariants. Subsequently 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. 2(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 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 Sec. 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 [7], 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, (Fig. 3) which we call "Anacin" and "Piggybank" respectively. The Anacin images consist of planes and the 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. In Fig. 3(b) we show the Bayesian likelihood ratio of these points for the entire sequence (continuous blue plot with 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 compared to the points in the first image, implying that the likelihood for the first image is 1.

[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.



(a) Anacin image (b) Anacin Likelihood (c) Piggybank image (d) Piggybank Likelihood

Figure 3: Likelihoods for sequences from the COIL database.

As may be observed, both the likelihood values stay close to 1 for most of the sequence and taper off towards the end of the sequence since here the vertical plane is almost parallel to the z-axis of the camera (i. e. viewing direction) resulting in pronounced perspective effects. The relative behaviour of the two plots is also interesting. As would be expected, 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. In Fig. 3(d), 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. 3(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 “affine” like as the effect of the non-planarity gets more pronounced. Thus the likelihood ratio falls off significantly. Thus, Fig. 3(d) tells us that the Piggybank is not a planar object while the points in one set of the Anacin sequence are coplanar. It also suggests that the second set of points in the Anacin sequence deviate less from coplanarity as compared to that of the Piggybank, thus confirming our notion of “affineness”. While we show the Bayesian likelihood here, the same results are obtained with the probabilistic invariant likelihood.

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. *Surveys*, 24(4):325–376, December 1992.

[2] A. Fitzgibbon and A. Zisserman. On affine invariant clustering and automatic cast listing in movies. In *European Conference on Computer Vision*, 2002.

[3] T. Leung, M. Burl, and P. Perona. Probabilistic affine invariants for recognition. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 1998.

[4] K. V. Mardia and Ian L. Dryden. *Statistical Shape Analysis*. John Wiley & Sons, 1998.

[5] J.L. Mundy and A. Zisserman. *Geometric Invariance in Computer Vision*. MIT Press, 1992.

[6] Isidore Rigoutsos and Robert Hummel. A Bayesian approach to model matching with geometric hashing. *Computer Vision and Image Understanding: CVIU*, 62(1):11–26, 1995.

[7] D. Sinclair and A. Blake. Quantitative planar region detection. *International Journal of Computer Vision*, 18(1):77–91, 1996.

[8] H.J. Wolfson and Y. Lamdan. Geometric hashing: A general and efficient model-based recognition scheme. In *International Conference on Computer Vision*, pages 238–249, 1988.