

# Multiple View 2D-3D Mutual Information Registration

M.E. Leventon, W.M. Wells III, W.E.L. Grimson

Massachusetts Institute of Technology Artificial Intelligence Laboratory

545 Technology Square, Cambridge, MA 02139

E-MAIL: leventon@ai.mit.edu

HOME PAGE: <http://www.ai.mit.edu/people/leventon>

## Abstract

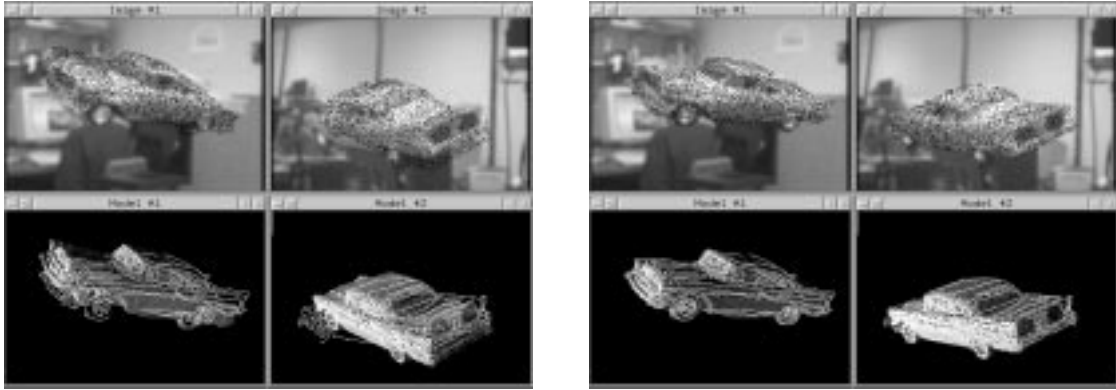
We present a method for finding the pose of an object in the world by registering a 3D model of the object to multiple images of the object taken from different positions by maximization of mutual information. Using multiple views of the object enables the registration process to converge on the three dimensional pose much more accurately than is possible from using just a single view. Since this method uses mutual information, the model of the object need only contain information about the shape of the object and need not know any details about other surface properties. Furthermore, this method is robust with respect to variations of illumination in the images. The method does not attempt to find any correspondences between pixels in the images, so the images of the object can be obtained from drastically different views and under different lighting conditions.

## 1 Introduction

Accurately computing the alignment of a 3D model of an object to an image of the object is an important problem in many computer vision applications. Many images of the same object can look very different, depending on object pose, lighting conditions, and other objects in the image. Therefore, the problem of computing the registration is a challenging one. One of the advantages of a mutual information approach to registration is that it is robust to many of the unknowns that can occur in an image, in-

cluding lighting conditions and occlusions [Viola and Wells, 1995]. Viola and Wells used this approach to compute a very accurate registration of a 3D model of an object to that object's position in the image plane. However, given that only one image was used to register the object, the registration found was not very accurate along the direction of the optical axis. Their error for registrations of a plastic skull were mostly under 2mm in the  $x$  and  $y$  dimension, but ranged from 5mm to almost 15mm in the  $z$  dimension [Viola and Wells, 1995]. A change in an object's  $x$  and  $y$  position is very noticeable in an image, while a shift in model position along the optical axis is difficult to see.

However, many applications require more than just an accurate registration in the image frame; an accurate 3D pose of the object is often needed. For example, a surgeon might want to register a patient's head to his or her internal CT/MR scan in order to very accurately position the patient for radiation therapy. During neurosurgery, the surgeon might want to point a trackable probe at some position inside the brain, and have the system display the 3D position of the probe in the CT/MR scan. Clearly, in these applications, being accurate in two dimensions is not enough. The motivation behind performing a mutual information registration using multiple views is to take advantage of the robustness in illumination variation and occlusion that mutual information offers, while also being able to accurately compute the pose of an object in three dimensions.



**Figure 1:** The first figure shows the initial random alignment of a test registration. The top two images show the views from the two cameras with randomly selected model points overlaid in red. The bottom two images show the model transformed by the pose and then projected into both image planes. The error associated with the initial pose is  $50.8mm$ . The second figure shows the result of using both views to register the model. The error in this final pose is  $3.1mm$ .

## 2 Previous Work

Much work has been done on the problem of registering a 3D model of an object to the world position of that object. Stereo methods of registration have the potential to improve the alignment along the optical axis, since depth information can be computed. However, stereo is susceptible to difficulty in finding correspondences between pixels in the images.

The difficulty in feature-based image registration lies in the problem of extracting common features between the model and the image. For example, edges extracted from an image can be due to albedo change, surface normal change, or illumination change (i.e. shadowing). The only types of edges that could be extracted from our shape model are edges due to change in surface normal. However, many objects have varying albedo and also shadow themselves, which will lead to many spurious edges in the image.

Fiducial registration involves manually picking corresponding points from the 3D model and the object. Accurately localizing these points is often difficult. For neurosurgical applications, Peters [Peters *et al.*, 1996] reports fiducial accuracy about an order magnitude worse than frame-based methods of registration, mainly due to the difficulty in accurately localizing the fiducial markers in both the internal scan and also on the patient.

## 3 Mutual Information Registration

In this section we review the basic method of alignment by maximization of mutual information, which has been described previously, [Viola and Wells, 1995] [Viola, 1995] [Wells *et al.*, 1995].

We seek an estimate of the transformation  $\hat{T}$  that aligns the model  $u$  and image  $v$  by maximizing their mutual information over the transformations  $T$ ,

$$\hat{T} = \arg \max_T I(u(x), v(T(x))) \quad .$$

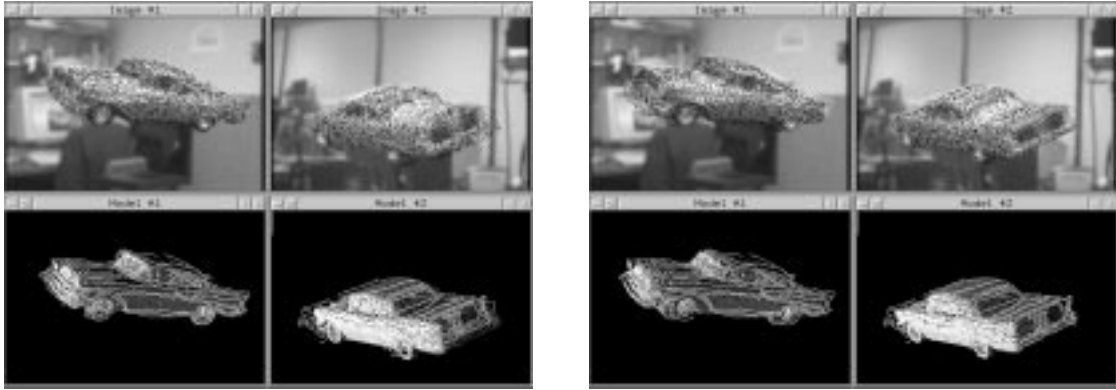
Here  $x$  is a random variable that ranges over visible surface patches in the model.

Mutual information is defined in terms of entropy in the following way:

$$I(u(x), v(T(x))) \equiv \quad (1) \\ H(u(x)) + H(v(T(x))) - H(u(x), v(T(x)))$$

.  $H(\cdot)$  is the entropy of a random variable, and is defined as  $H(x) \equiv - \int p(x) \ln p(x) dx$ . The joint entropy of two random variables  $x$  and  $y$  is  $H(x, y) \equiv - \int p(x, y) \ln p(x, y) dx dy$ . Entropy can be interpreted as a measure of uncertainty, variability, or complexity.

Information has three components. The first term on the right in Equation 1 is the entropy in the model. It is not a function of  $T$ . The second



**Figure 2:** The first figure shows the result of just using the *first* image to register, and the second figure shows the result of only using the *second* image to register. Notice that in both cases, the registration is good in the image plane of the view that was used but is off in the other view.

term is the entropy of the part of the image into which the model projects. It encourages transformations that project  $u$  into complex parts of  $v$ . The third term, the (negative) joint entropy of  $u$  and  $v$ , takes on large values if  $u$  and  $v$  are functionally related. It encourages transformations where  $u$  explains  $v$  well. Together the last two terms identify transformations that find complexity and explain it well. This is the essence of mutual information.

In [Viola and Wells, 1995] and [Viola, 1995] a stochastic gradient descent method was used to seek local maxima of the mutual information criterion, and [Wells *et al.*, 1995] described a gradient method that uses histograms to approximate entropies and their derivatives. The latter method was used in the work reported here.

## 4 Multiple View Registration

The goal of the multiple view 2D-3D mutual information registration approach is to find the pose of the model that best describes all the images of the object. The algorithm is very similar to that which is described in [Viola and Wells, 1995]. To apply the mutual information registration technique to multiple views, we perform a single-view registration “step” for each view in turn.

The algorithm requires a point/normal model  $M$  of the object, and  $n$  images of that object

$\{I_1, \dots, I_n\}$ . Additionally, the relative poses (positions and orientations) of the  $n$  cameras must be known. Let  $T_j \in \{T_1, \dots, T_n\}$  be the transformation that takes a point in world coordinates into Camera  $j$ ’s coordinates. (Assume, for simplicity, that  $T_1$  is the identity, so world coordinates are the same as Camera 1 coordinates.) Finally, the algorithm requires an initial pose  $P_0$  from which to perform the gradient ascent.

At each iteration, for each view, the algorithm updates the pose in the direction of the gradient of mutual information for that view.

```

P ← P0
For each iteration  $i$ , ( $i = 1, 2, \dots$ )
  For each view  $j$ , ( $j = 1, \dots, n$ )
    Define:
      Pj ← PM
      Mj ← TjPj
    Compute:
      ΔPj ← ∇MI(Mj, Ij)
    Let:
      D = translation(Pj)
      R = quaternion_rotation(Pj)
      d = λd × translation(ΔPj)
      r = scale_rotation
          (quaternion_rotation(ΔPj), λr)
    Update:
      P'j(·) ← r(R(·)) + D + d
      P ← Tj-1P'j

```

The functions  $translation(P)$  and  $quaternion\_rotation(P)$  extract the translational and rota-

tional components of the pose  $P$  respectively. A rotation  $r$  can be represented as a rotation angle  $\theta$  about some unit vector  $v$  (so  $r = \langle \theta, v \rangle$ ). The function *scale\_rotation* scales the rotation angle  $\theta$  such that  $scale\_rotation(\langle \theta, v \rangle, \lambda_r) = \langle \lambda_r \theta, v \rangle$ . Note that in the first update step,  $P'_j$  is set to the result of composing  $P_j$  with a scaled  $\Delta P_j$

## 5 Results

In this section, we present the results of running multiple registration experiments using two views of a model car. A 3D point-normal model of the car was derived from a computed tomography (CT) scan. Two cameras were placed approximately 1.5m apart aimed at the model car that is 0.5m in length and positioned about 1.0m away from the two cameras. The images of the car taken from the two cameras are shown in figures 1. A “correct” pose was determined by manually aligning the 3D model in both image frames. This pose will be used as the ground truth for the registration experiments.

In order to evaluate a pose that is returned by the registration algorithm, it is necessary to define a distance or error metric for poses. The error metric we used is defined as follows:

$$E_{3D}(P|P^*, M) = \max_{q \in M} \|Pq - P^*q\|$$

The error in pose, given the “correct” pose  $P^*$  and the model  $M$ , is the maximum distance between corresponding model points under the two transformations.

### 5.1 Results on One Example Trial

Starting with a initial random pose with an error of 50.8mm, after 200 iterations of the algorithm, using both views, the final pose error is 3.1mm. Figure 1 shows the initial pose and the final pose. As a comparative measure, the algorithm was run twice more, once using only the first view, and a second time using only the second view, again for 200 iterations starting from the same initial pose. The pose errors for these single-view registrations were 17.3mm and 26.8mm respectively. Figure 2 shows the results of these registrations. Notice that for

both registrations, the algorithm did a good job of locking down the registration in the  $x$  and  $y$  direction of the view it processed, but did not register the model very accurately in the  $z$  direction, or the direction of the optical axis. In both cases, this error in registration is noticeable in the other view of the car that was not used in the registration.

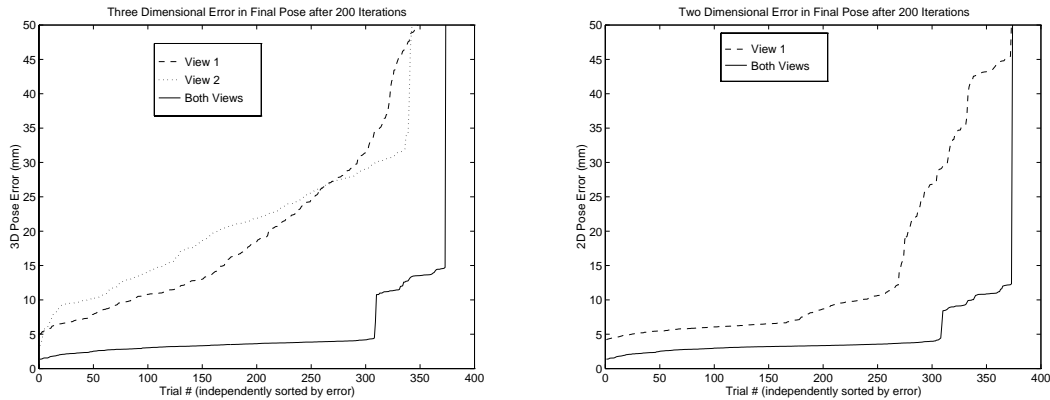
### 5.2 Results Over Many Trials

The two-view mutual information registration algorithm was run on the same car images (in figure 1) with 400 random starting model poses. Each random initial pose was within  $\pm 15mm$  and  $\pm 10^\circ$  in each dimension of the “right” pose. For each random pose, the algorithm was run for 200 iterations, once using only the first view, once using only the second view, and once using both views. The graph in Figure 3 shows the results from these 400 trials, sorted by error. As seen in the graph, the two view registration process aligned the model within 3.5mm of the correct pose slightly over 80% of the time. When the algorithm had only one view to work with, the registration error was significantly greater, and from the graph, it is even difficult to see when it converged near the correct solution and when it found an incorrect pose.

However, one of the first observations and the main motivation for using multiple views was that a single view has very limited depth information, and *any* registration algorithm would have difficulty accurately registering an object along the optical axis. Thus, comparing the single-view and two-view algorithms in this way — by using three dimensional distance — is not really a fair comparison. Therefore, we define the following two dimensional error metric:

$$E_{2D}(P|P^*, M) = \max_{q \in M} \|F_P(Pq) - F_P(P^*q)\|$$

where  $F_P$  takes a point in the 3D coordinates of a camera and projects it into 2D image coordinates.  $E_{2D}$ , unlike  $E_{3D}$ , only considers the error in the image plane, and ignores any error along the optical axis. Thus,  $E_{2D}$  would seem to be a more “fair” error measurement to use when comparing a multiple-view registration to a single-view registration.



**Figure 3:** The first graph illustrates the 3D error in pose after 200 iterations over the 400 registrations with random initial poses. The second graph illustrates the 2D pose error. In this graph, any error along the optical axis is ignored.

Figure 3 shows the results of the 400 trials using the two dimensional error metric. Notice that now it is much more obvious when the single-view algorithm converged near the correct solution. However, the two-view approach still performs significantly better in a few different ways. First, the pose error of the two-view algorithm is still much less that of the single-view algorithm, usually by a factor of two. This implies that using two views not only improves the registration along the optical axis, but also yields a better registration in the image plane.

In addition to returning more accurate registrations, the two-view method also seems to have a much larger region of convergence. Of the 400 trials, the two-view approach registered well 80% of the time, registered reasonably about 15% of the time, and diverged the remaining 5%. The single view method registered well only about 50% of the time, registered reasonably about 25% of the time, and diverged about 25% of the time. Thus, using two views seems to drastically improve the registration of the model to the position of the 3D object in the world.

One disadvantage of using multiple views is that it can be slower by a factor of  $n$  for  $n$  views. Generally, though, the registration actually converges much faster (and is more likely to converge), so the slowdown may not be significant. Another disadvantage of using  $n$  views is that it requires the calibration of  $n$  cameras. In some applications, it might be difficult to precisely

calibrate the  $n$  cameras.

## References

- [Grimson *et al.*, 1996] W.E.L. Grimson, G.J. Ettinger, S.J. White, T. Lozano-Pérez, W.M. Wells III, and R. Kikinis. “An Automatic Registration Method for Frameless Stereotaxy, Image Guided Surgery, and Enhanced Reality Visualization”. *IEEE TMI*, **15**(2):129–140, April 1996.
- [Horn, 1987] B. Horn. “Closed-form Solution of Absolute Orientation Using Unit Quaternions”. *JOSA A*, **4**:629–642, April 1987.
- [Peters *et al.*, 1996] T. Peters, B. Davey, P. Munger, R. Comeau, A. Evans, A. Olivier. “Three-Dimensional Multimodal Image-Guidance for Neurosurgery”. *IEEE TMI*, **15**(2):121–128, April 1996.
- [Viola and Wells, 1995] P.A. Viola, W.M. Wells III. “Alignment by Maximization of Mutual Information”. In *International Conference on Computer Vision*, June 1995.
- [Viola, 1995] P.A. Viola. PhD thesis. MIT Department Electrical Engineering and Computer Science, Cambridge, Mass., 1995.
- [Wells *et al.*, 1995] W. Wells III, M. Halle, R. Kikinis, P. Viola. “Alignment and Tracking using Graphics Hardware”. In *Proceedings of the Image Understanding Workshop*, Feb. 1996.