Local Cross-Modality Image Alignment Using Unsupervised Learning

Öjvind Bernander and Christof Koch

Computation and Neural Systems Program, 216-76
California Institute of Technology, Pasadena, Ca 91125

Abstract

We propose a method for automatically aligning images with local distortions from different sensors, using real images instead of calibration objects. The algorithm has three components. First, we extract intensity discontinuities, because this is a feature that is likely to show up across modalities. Second, we use a correlation scheme that averages over time rather than space, for high precision. Third, we propose an architecture and a learning scheme that learn the correlation surfaces over time and implement the image coordinate transform.

Introduction and problem definition

Fusion, or integration, of information from different sensors is believed to facilitate object recognition. The sensors may be of different modalities, e.g. video, infra-red or laser range cameras. Before fusion can occur, however, the images must be properly aligned. Thus the problem is defined: *given two cameras at two positions with overlapping fields-of-view, find the coordinate transform that will align the overlapping portions.* We represent the transform using a *shift field*, a vector field sharing some formal properties with the optical flow field. Note that the misalignment generally will vary across the image, due to rotation, zoom, and local distortions. The simplest approach is to use special calibration objects, e.g. *hot corners*, to produce calibration images, and then interpolate between these points. This approach presents problems in remote-control or autonomous situations (e.g. for a Mars rover or for biological visual systems) when calibration objects are not likely to be at hand. This is the motivation for developing an algorithm that achieves image alignment using natural images.

A correlation scheme for image alignment

One problem with using different modalities is that there is expected to be little correlation between intensity values, since reflectance, temperature, and distance correlate only to a very small degree. We therefore need to extract features that are likely to show up in any modality. Edges, or intensity discontinuities, have this property, since they often arise at object boundaries. This was recognized by Barniv and Casasent [1], who studied *full-image* correlations between images that were captured with the same camera, but with different filters. Correlating edge maps yielded a clear peak, the position of which gives any shift between the two images.

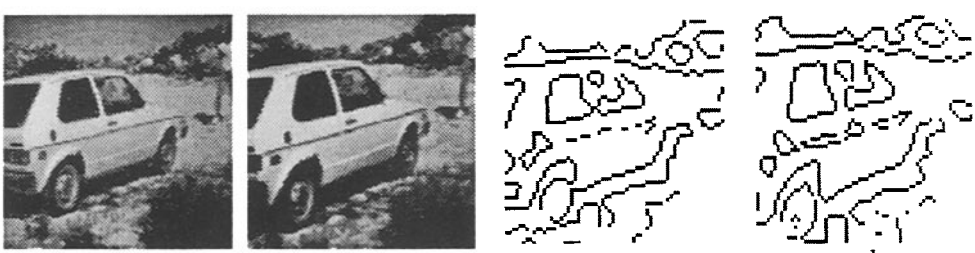

Figure 1: Sample image pairs.
(a) and (b) depict the same scene, but are captured using two video cameras of different makes. (c) and (d) show the thresholded zero-crossings of (a), after filtering with the Laplacian of a Gaussian.

In the general case, however, the two sensors do not have identical fields-of-view and resolutions. While a full-image correlation will work for pure translations, it will not work for rotations and zooms (different resolutions). To remedy this we average in time rather than space. Let us define the *atomic correlation function* $\mathbf{a}_{i,j} = m^{(1)}_{i,j} * \mathbf{m}^{(2)}_{i,j}$, where $m^{(1)}_{i,j}$ is the pixel value at (i,j) in image 1 and $\mathbf{m}^{(2)}_{i,j}$ is a patch of pixels in image 2. $\mathbf{a}_{i,j}$ defines a correlation surface that is local in both space and time. Averaging $\mathbf{a}_{i,j}$ over the whole image, we get a full-image correlation. Averaging $\mathbf{a}_{i,j}$ over a patch gives patch correlation. We average $\mathbf{a}_{i,j}$ in time over a set of image pairs, sometimes in combination with a small amount of spatial averaging over 2x2 or 3x3 squares, in order to reduce the number of image pairs we need. Hence we get one correlation surface at every pixel position in image 1. We use a 2-D parabolic fit to find the position of the peak of the correlation surface, using sub-pixel accuracy. This position vector gives the local misalignment and the set of all position vectors defines a *shift field*, which gives us the desired coordinate transform.

To generate a database of 180 video image pairs, we directed two cameras of different makes towards a screen upon which vacation slides were projected. The digitized images were filtered with the Laplacian of a Gaussian, and edges were marked at the zero-crossings. A sample image pair is shown in figure 1. Figure 2(a-b) shows sample correlation surfaces and 2(c) the shift field displayed as a needle diagram. One camera was zoomed-in compared to the other, which clearly shows in figures 1 and 2(c). The average error in peak position was very low. For auto-alignment, i.e. when images 1 and 2 were identical and the true misalignment was known, the average error in peak position was less than 0.1 pixels. Figure 3 shows how the average error in peak position varies with the number of images used and with the amount of spatial averaging. For cross-alignment, i.e. when images 1 and 2 were different and the true misalignment was not known, we estimated the standard deviation to $\sigma = 0.2$ pixels. To assess the robustness to noise we generated artificial Mondrian images and added various amounts of salt-and-pepper noise. We found that even at very low signal to noise ratios (SNR = 1.0) the average position error was less than 0.2 pixels and the width of the correlation peak increased by 50%.

An architecture for learning and implementing the coordinate transform

One way to implement the shift would be to use the setup depicted in figure 4. A patch $\mathbf{m}^{(2)}$ of neurons in image 2 project via a set of weights $\mathbf{w}$ (referred to as a receptive field) to a neuron with output $y = \mathbf{m}^{(2)} \cdot \mathbf{w}$. If only one component of $\mathbf{w}$ is non-zero, the corresponding component of $\mathbf{m}^{(2)}$ will be shifted into y. We can approximate the ideal receptive field $\mathbf{w}$ with the correlation surface. A similar approach was suggested by [4]. With $\mathbf{a}$ defined as above, i.e. $\mathbf{a}_{i,j} = m^{(1)}_{i,j} * \mathbf{m}^{(2)}_{i,j}$, the learning rule $\dot{\mathbf{w}} = \alpha\mathbf{a} + \beta y\mathbf{w}$ (α and β positive constants) will converge to $\mathbf{w}(t = \infty) \propto \mathbf{a}_{average}$ [3],

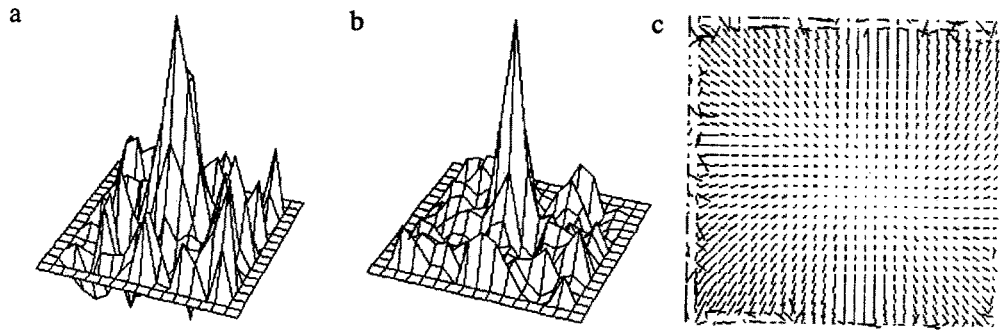

Figure 2: Correlation surfaces and shift field.

Using a set of 180 edge image pairs, an example of which is given in figure 1 (c-d), the time-averaged correlation was calculated at every pixel. (a) A typical correlation surface at a sample pixel position. (b) The average correlation surface of four neighbors. The peak is somewhat more pronounced, and its position better defined. (c) Shift field. Parabolic fits were used to find the peak positions which are represented as needles in the diagram. The zoom effect is obvious. For clarity only every fourth needle is shown.

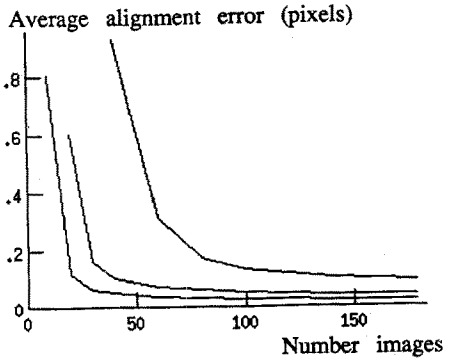

Figure 3: Accuracy of alignment.
The alignment error decreases rapidly as more images are used. The curves represent different amounts of spatial averaging. From top to bottom, no averaging, 2x2 averaging, 3x3 averaging.

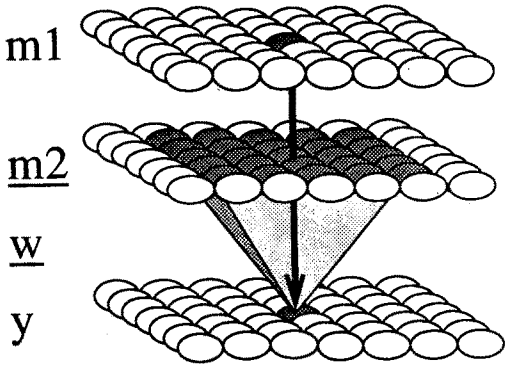

Figure 4: Learning Architecture. See text for details.

i.e. the time-averaged correlation surface. We used the same image set as in the previous section to test this learning algorithm, presenting the images in random order. Starting from random weights or a peak in the wrong position, we get convergence after 300–500 iterations.

If this algorithm runs continuously on an autonomous vision system, it would rapidly adapt to changes in camera positions or other distortions. Such dynamic realignment has been shown to occur in the barn owl optic tectum [2]. A VLSI implementation of the algorithm would be useful for integrated early vision modules now under development.

Further discussion

Our problem definition translates to finding correspondences between image pairs. This is related to the problem of binocular stereo and image motion, and algorithms developed in these areas could conceivably be used. However, the problem to be solved *is not* analogous, and our algorithm has two advantages. First, stereoscopic effect are a major source of error, since a close object would be interpreted as a local distortion in the imaging equipment. Averaging over several image pairs is a necessity and reduces stereoscopic errors at the cost of a wider peak. Second, time-averaging over many images yields very high precision and allows for a straightforward implementation of the learning algorithm as described in the previous section.

We would like to thank the Hughes Aircraft AI Center for partial support for this research.

References

[1] Barniv, Y. and Casasent, D. (1981) *Multisensor image registration: experimental verification*, SPIE 292: Processing of images and data from optical sensors, 160-171.

[2] Knudsen, E.I. (1983) *Early auditory experience aligns the auditory map of space in the optic tectum of the barn owl* , Science 222, 939-942.

[3] Kohonen, T. (1984) *Self-organization and associative memory* , Springer-Verlag, Berlin.

[4] Pearson, J.C., Sullivan, W.E., Gelfand, J.J., Peterson, R.M. (1987) *A computational map approach to sensory fusion* , AOG/AAAIC Proc. Joint conf. on merging tomorrow's technologies with defense readiness requirements