Pose-invariant 3d object recognition using linear combination of 2d views and evolutionary optimisation

Vasileios Zografos and Bernard F. Buxton
Department of Computer Science,
University College London,
Malet Place, London, WC1E 6BT
{v.zografos, b.buxton}@cs.ucl.ac.uk

Abstract

In this work, we present a method for model-based recognition of 3d objects from a small number of 2d intensity images taken from nearby, but otherwise arbitrary viewpoints. Our method works by linearly combining images from two (or more) viewpoints of a 3d object to synthesise novel views of the object. The object is recognised in a target image by matching to such a synthesised, novel view. All that is required is the recovery of the linear combination parameters, and since we are working directly with pixel intensities, we suggest searching the parameter space using an evolutionary optimisation algorithm in order to efficiently recover the optimal parameters and thus recognise the object in the scene.

1 Introduction

Object recognition is one of the most important and basic problems in computer vision and, for this reason, it has been studied extensively resulting in a plethora of publications and a variety of different approaches[1] aiming to solve this problem. Nevertheless accurate, robust and efficient solutions remain elusive because of the inherent difficulties when dealing in particular with 3d objects that may be seen from a variety of viewpoints. Variations in geometry, photometry and viewing angle, noise, occlusions and incomplete data are some of the problems with which object recognition systems are faced.

In this paper, we will address a particular kind of extrinsic variations: variations of the image due to changes in the viewpoint from which the object is seen. Traditionally, methods that aimed to solve the recognition problem for objects with varying pose relied on an explicit 3d model of the object, generating 2d projections from that model and comparing them with the scene image. Such was the work by Lee and Ragnarath [13]. Although 3d methods can be quite accurate when dealing with pose variations, generating a 3d model can be a complex process and require the use of specialised hardware. Other methods [12, 3] have thus tried to capture the viewpoint variability by using multiple views of the object from different angles, covering a portion of, or the entirety of, the view sphere. If the coverage is dense these methods require capture and storage of a vast number of views for each object of interest. Quite recently, new methods have been introduced that try to alleviate the need for many views while still working directly with 2d images.They are called *view-based* methods and represent an object as a collection of a small number of 2d views. Their advantage is that they do not require construction of a 3d model while keeping the number of required stored views to a minimum. Prime examples are the works by Bebis et al. [1], Turk and Pentland [21] and Cootes et al. [6].

Our proposed method is a view-based approach working directly with pixel values and thus avoids the need for low-level feature extraction and solution of the correspondence problem such as in [1]. As a result, our model is easy to construct and use, and is general enough to be applied across a variety of recognition problems. The disadvantage is that it may also be sensitive to illumination changes, occlusions and intrinsic [7] shape variations. We adopt a “generate and test” approach using an evolutionary algorithm to recover the optimal LCV coefficients that synthesise a novel image, which is as similar as possible to the target image. If the similarity (usually the cross-correlation coefficient) between the synthesised and the target images is above some threshold then an object is determined to be present in the scene and its location and pose are defined (at least in part) by the LCV coefficients.

In the next section we introduce the LCV and explain how it is possible to use it to synthesise realistic images

[1] For a comprehensive review of object recognition methods and deformable templates in particular, see [10, 16, 23, 2].

from a range of viewpoints. In section 3 we present our 3d object recognition paradigm which incorporates the LCV and the optimisation algorithm, and in section 4 we show some experimental results of our approach on synthetic and real imagery. Finally, we conclude in section 5 with a critical evaluation of our method and suggestion how it could be further improved in the future.

2 Linear combination of views

LCV is a technique which belongs in the general theory of the tri- and multi-focal tensors, or Algebraic Function of View (AFoV) [18] and provides a way of dealing with variations in an object's pose due to viewpoint changes. This theory is based on the observation that the set of possible images of an object undergoing 3d rigid transformations and scaling is, under most (i.e. affine) imaging conditions, to a good approximation embedded in a linear space spanned by a small number of 2d images. It therefore follows that the variety of 2d views depicting an object can be represented by a combination of a small number of 2d *basis views* of the object.

Ullman and Basri [22] were the first to show how line drawings or edge map images of novel views of a 3d object could be generated via a linear combination of similar 2d basis views. More specifically, they showed that under the assumption of orthographic projection and 3d rigid transformations, 2 views are sufficient to represent any novel view of a polygonal object from the same aspect. The proof may easily be extended to any affine imaging condition. Thus, to a good approximation, given two images of an object from different (basis) views I' and I'' with corresponding image coordinates (x', y') and (x'', y''), we can represent any point (x, y) in a novel view I according to, for example:

$$\begin{array}{r} x = a_0 + a_1 x' + a_2 y' + a_3 x'' \\ y = b_0 + b_1 x' + b_2 y' + b_3 x'' \end{array} . \tag{1}$$

The novel view is reconstructed from the above two equations given a set of valid coefficients (a_i, b_j). Provided we have at least 4 corresponding "landmark" points in all three images (I, I', I'') we can estimate the coefficients (a_i, b_j) by using a standard least squares approach[2]. Several others have taken this concept further from its initial application to line images and edge maps to real images [11, 9, 15, 1] .

Such results suggest that it is possible to use LCV for object recognition in that novel views of an object can be recognised by matching them to a combination of stored, basis views of the object. The main difficulty in applying this idea within a pixel-based approach is the selection of the LCV coefficients (a_i, b_j). In particular, as described in the next section, synthesis of an image of a novel view from the images of the basis views, although straightforward, is a non-linear and non-invertible process.

2.1 Image synthesis

To synthesise a single, novel image using LCV and two views we first need to determine its geometry from the landmark points. In principle we can do so by using (1) and n corresponding landmark points (where $n \geqslant 4$), and solving the resulting system of linear equations in a least squares sense. This is straightforward if we know, can detect, or predict the landmark points in image I. Such methods may therefore be useful for image coding and for synthesis of novel views of a known object [11, 9]. For pixel-based object recognition in which we wish to avoid feature detection a direct solution is not possible, but we instead use a powerful optimisation algorithm to search for and recover the LCV coefficients for the synthesis.

Given the geometry of the novel image I, in a pixel-based approach we need to synthesise its appearance (colour, texture and so on) in terms of the basis images I' and I''. Since we are not concerned here with creation of a database of basis views of the objects of interest, we may suppose that a sparse set of corresponding landmark points $(x'(j), y'(j))$ and $(x''(j), y''(j))$ may be chosen manually and offline in images I' and I'' respectively and used to triangulate the images in a consistent manner. An illustration of the above can be seen in Fig. 1.

Given a set of hypothesised landmark points $(x(j), y(j))$ in the target image we can, then to a good approximation, synthesise the target image I as described in [5, 7, 11] from a weighted linear combination:

$$I(x, y) = w' I'(x', y') + w'' I''(x'', y'') + \epsilon(x, y), \tag{2}$$

in which the weights w' and w'' my be calculated from the LCV coefficients. Essentially this relies on the fact that, in addition to the multi-view image geometry being to a good approximation affine, the photometry is to a good approximation affine or linear [17]. The synthesis essentially warps and blends images I' and I'' to produce I. It is important to note therefore that (2) applies at all points (pixels) (x, y), (x', y') and (x'', y'')in images I, I' and I'' with the dense correspondence defined by means of the LCV equations (1) and a series of piecewise linear mappings [8] within each triangle of the basis images. If (x', y') and (x'', y'') do not correspond precisely to pixel values, bilinear interpolation is used [9, 11]. The same idea may be extended to colour

[2] It has also been shown that more general and in particular a more symmetrical set of equation involving all the basis view co-ordinates may be used, though in general such equations are over-complete. The general solution is to express the LCV in terms of an affine tri-focal tensor [7], though for our purposes where the changes of viewpoint are mainly in a single direction between the basis views, which may be used to define the x, x' and x'' axes, (1) suffices.

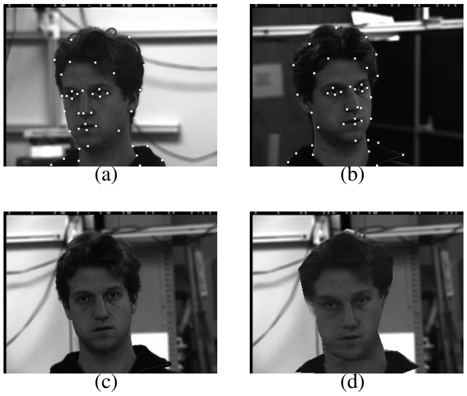
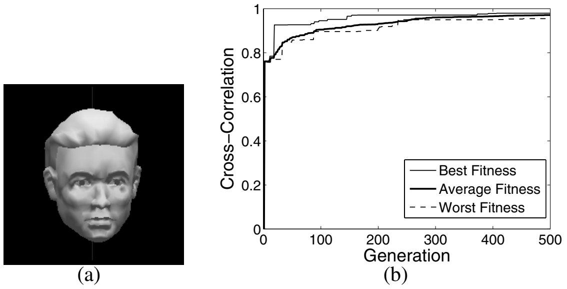

(a)
(b)
(c)
(d)

Figure 1. Example of real data from the CMU PIE database. The two basis views (a) and (b) and the target image (c). The synthesised image (d) is at the correct pose identified by our algorithm.

images by treating each spectral band as a luminance component (e.g. I_R, I_G, I_B).

3 The recognition system

In principle using the LCV for object recognition is easy. All we have to do is find the LCV coefficients in an equation such as (1) which will optimise the sum of squared errors ϵ from (2) and check if it small enough to enable us to say our synthesised and target images match.



(a)
(b)

Figure 2. Example of a synthetic image used for testing (a). The average test results are shown in (b).

3.1 Template matching

The first component of our system is the two stored basis views I' and I''. These are rectangular bitmap images that contain gray-scale, pixel information of the object without any additional background data. The images are obtained from basis views chosen, as indicated earlier, so that the viewpoint from which the target image I is taken lies on the view sphere between or almost between the basis views from which I' and I'' are taken. It is important not to choose a very wide angle between the basis views since this can lead to I' and I'' belonging to different aspects of the object and thus to landmark points being occluded[3].

Having selected the two basis views, we pick a number of corresponding landmark points in particular lying on discontinuity boundaries, edges and other prominent features. When the appropriate number of landmarks have been selected we use constrained Delaunay triangulation to produce consistent and corresponding triangular meshes of all the images. The above processes may be carried out during an offline training stage and are not examined here. The recognition system involves choosing the appropriate LCV coefficients (a_i, b_j), synthesising an image I_s and comparing it with the target image I, using some similarity or dissimilarity metric. Since we compare the two images over all pixels (i.e. over both foreground and background), either a dissimilarity metric such as the sum of squared differences (SSD) or a similarity measure such as the cross-correlation coefficient $c(I, I_s)$ may be used. We have used the latter because when applied to the whole image it is invariant to affine photometric transformations [4]. The choice of LCV coefficients is determined by maximising the cross-correlation coefficient. Essentially we are proposing a flexible template matching system, in which the template is allowed to deform in the LCV space until it matches the target image. The only component that affects the deforming template is the match or miss-match between the target and scene images.

3.2 Optimisation

To find the LCV coefficients we need to search a high-dimensional parameter space using an efficient optimisation algorithm. For this purpose, we have chosen a recent evolutionary, population-based optimisation algorithm that works on real-valued coded individuals and is capable of handling non-differentiable, nonlinear and multi-modal objective functions. It is called Differential Evolution (DE) and was introduced by [20]. Briefly, DE works by adding the weighted difference between two randomly chosen population vectors to a third vector, and the fitness resultant is compared with that of another individual from the current population. In this way, DE can deduce from the distances between the population vectors where a better solu-

[3] It is still quite possible to synthesise novel images at wider angles and remove any self-occluded triangles. Although we do not address this problem here, see [9].

tion might lie, thereby making DE self-organising. We have chosen DE because it is very easy to set-up and use, especially in a template matching scenario where we are using a mixture of discrete and continuous parameters. In addition, it is efficient in searching high-dimensional spaces and is capable of finding promising basins of attraction [4] early in the optimisation process without the need for good initialisation.

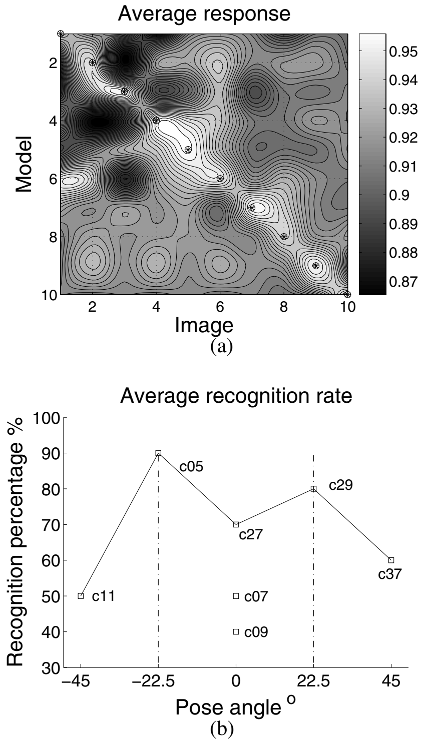


Figure 3. A colormap plot (a) showing the entries in the confusion array averaged across pose. (b) shows the recognition rate across pose angle for all models.

4 Experimental results

We performed a number of experiments on synthetic and real images under varying camera angle. The synthetic images were generated by taking 2d snapshots of a 3d object (a human head model) in front of a black background (see Fig. 2(a)). Landmarks where manually selected amongst the vertices of the 3d object and their projected positions were automatically calculated in the 2d images. For the real images, we used pose variation subsets from the CMU PIE [19] database, making sure the chosen landmarks where visible in both basis views (see Fig. 1). In the synthetic dataset, the pose angles where chosen mainly between $\pm 14^o$ about the vertical and $\pm 10^o$ about the horizontal axes. The majority of the target views lay on the view-sphere between the basis views, but in a few examples the system had to extrapolate the data in order to recover the optimal coefficients. In total, we ran 10 synthetic experiments and the results are illustrated in Fig. 2.

These results are very encouraging with the majority of the experiments converging to the correct solution with a cross-correlation of > 0.97. Only cases in which the target viewpoint was far from the line in view space between the basis views failed to converge. In such cases, the LCV could not synthesise the target view accurately indicating the need to use more than two basis views in order to better represent that portion of the view-sphere.

For the real image experiments on the CMU PIE database, we constructed LCV models from 10 individuals using as basis views, the left and right images (*c29, c05*) of each individual at the natural expression. The face once synthesised was then superimposed onto the background which is given separately in the database and the resulting image was compared with a test view. Comparisons were carried out against the images of the 10 individuals in the database while attempting to detect poses from $-45^o, -22.5^o, 0^o, 22.5^0, 45^o$ about the vertical and a limited range about the horizontal axes (images *c09* and *c07*).

In total we carried out 700 experiments across pose and constructed a $10 \times 10 \times 7$ "confusion array" of model×image×pose. Each 10×10 pose-slice of this array contains information about the recognition responses (cross-correlation) of our tests, the highest being along the main diagonal, where each individual's model is correctly matched to that individual's image. The recognition response should fall off when comparing a specific model with images of other individuals. This behaviour, averaged across pose can be seen in Fig. 3(a) and the pose-dependent recognition rate (averaged across the 10 models) in Fig. 3(b). The results are quite pleasing with the correct pose identified the majority of times when the target view was between the basis views ($\pm 22.5^o$) as no extrapolation is required. The recognition rate falls off when the target examples lay outside that range, ($\pm 45^o$) and for images $c09$ and $c07$. It is possible to increase this considerably using more basis views during the modelling stage. In addition, constructing more models and carrying out additional tests would provide more accurate recognition rates, especially for the fronal image *(c27)* at 0^o.

5 Conclusion

We have shown how the linear combination of views (LCV) method may be used in view-based object recognition. Our approach involves synthesising intensity images using LCV and comparing them to the target, scene image using a similarity metric. The optimal LCV coefficients for the synthesis are recovered by an evolutionary algorithm, differential evolution [20]. Experiments on both synthetic and real data demonstrate that the method works well for pose variations especially those where the target view lies between, or almost between the basis views. DE plays an important role in our method, by searching efficiently the high-dimensional, LCV space. Such solutions can narrow the search space to a promising basin of attraction within which a local optimisation method may be used for finding an accurate solution.

Further work is required, however. In particular, we would like to reformulate (1) by using the affine tri-focal tensor and introducing the appropriate constraints in the LCV mapping process. Formulating (1) in term of individual 3d transforms might also help bound the range of the LCV coefficients and make the selection process more intuitive. Furthermore, we would like to introduce probabilistic weights on the coefficients as prior information about the range of likely views and formulate a Bayesian inference mechanism. This, we believe, will greatly aid the recognition process. At this stage we have only addressed extrinsic, viewpoint variations, but we have indicated how it should be possible to include intrinsic, shape variations (see for example [7] and lighting variations on the image pixels. In addition, more experiments are needed in order to evaluate the performance of our method against public available datasets (e.g. [14]) and against other, competing methods designed to solve the same problem, such as the Active Appearance Models [6].

References

[1] G. Bebis, S. Louis, T. Varol, and A. Yfantis. Genetic object recognition using combinations of views. *IEEE Transactions on Evolutionary Computation*, 6(2):132–146, April 2002.

[2] P. J. Besl and R. C. Jain. Three-dimensional object recognition. *ACM Computing Surveys (CSUR)*, 17:75–145, 1985.

[3] D. J. Beymer. Face recognition under varying pose. In *Proc. IEEE Conf. CVPR*, pages 756–761, 1994.

[4] B. Buxton and V. Zografos. Flexible template and model matching using image intensity. In *Proceedings Digital Image Computing: Techniques and Applications (DICTA)*, 2005.

[5] B. F. Buxton, Z. Shafi, and J. Gilby. Evaluation of the construction of novel views by a combination of basis views. In *Proc. IX European Signal Processing Conference (EUSIPCO-98)*, Rhodes, Greece, 1998.

[6] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *IEEE Pattern Analysis and Machine Intelligence*, 23:681–685, 2001.

[7] M. B. Dias and B. F. Buxton. Implicit, view invariant, linear flexible shape modelling. *Pattern Recognition Letters*, 26(4):433–447, 2005.

[8] A. Goshtasby. Piecewise linear mapping functions for image registration. 19(6):459–466, 1986.

[9] M. E. Hansard and B. F. Buxton. Parametric view-synthesis. *In Proc. 6th ECCV*, 1:191–202, 2000.

[10] A. K. Jain, Y. Zhong, and M.-P. Dubuisson-Jolly. Deformable template models: A review. *Signal Processing*, 71(2):109–129, 1998.

[11] I. Koufakis and B. F. Buxton. Very low bit-rate face video compression using linear combination of 2dfaceviews and principal components analysis. *Image and Vision Computing*, 17:1031–1051, 1998.

[12] Y. Lamdan, J. Schwartz, and H. Wolfson. On recognition of 3d objects from 2d images. *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1407–1413, 1988.

[13] M. W. Lee and S. Ranganath. Pose-invariant face recognition using a 3d deformable model. *Pattern Recognition*, 36:1835–1846, 2003.

[14] S. A. Nene, S. K. Nayar, and H. Murase. Columbia Object Image Library (COIL-100). Technical Report CUCS-006-96, Columbia University, 1996.

[15] G. Peters and C. von der Malsburg. View reconstruction by linear combination of sample views. *In Proc. British Machine Vision Conference BMVC 2001*, 1:223–232, 2001.

[16] A. R. Pope. Model-based object recognition. a survey of recent research. Technical Report 94-04, 1994.

[17] A. Shashua. *Geometry and Photometry in 3D Visual Recognition.* PhD thesis, Massachusetts Institute of Technology, November 1992.

[18] A. Shashua. Algebraic functions for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):779–789, 1995.

[19] T. Sim, S. Baker, and M. Bsat. The CMU pose, illumination and expression (PIE) database. In *Proc. of the 5th IEEE international conference on automatic face and gesture recognition*, 2002.

[20] R. Storn and K. V. Price. Differential evolution - a simple and efficient heuristic for global optimization overcontinuous spaces. *Journal of Global Optimization*, 11(4):341–359, December 1997.

[21] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[22] S. Ullman and R. Basri. Recognition by linear combinations of models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):992–1006, October 1991.

[23] M.-H. Yang, D. J. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.