# Combining Single View Recognition and Multiple View Stereo For Architectural Scenes

A. R. Dick
Dept. Engineering
Univ. Cambridge
Cambridge, UK
ard28@cam.ac.uk

P. H. S. Torr
Microsoft Research
St. George House
Cambridge, UK
philtorr@microsoft.com

S. J. Ruffle
Dept. Architecture
Univ. Cambridge
Cambridge, UK
sjr56@cam.ac.uk

R. Cipolla
Dept. Engineering
Univ. Cambridge
Cambridge, UK
cipolla@eng.cam.ac.uk

## Abstract

*This paper describes a structure from motion and recognition paradigm for generating 3D models from 2D sets of images. In particular we consider the domain of architectural photographs. A model based approach is adopted with the architectural model built from a "Lego kit" of parameterised parts. The approach taken is different from traditional stereo or shape from X approaches in that identification of the parameterised components (such as windows, doors, buttresses etc) from one image is combined with parallax information in order to generate the 3D model. This model based approach has two main benefits: first, it allows the inference of shape and texture where the evidence from the images is weak; and second, it recovers not only geometry and texture but also an interpretation of the model, which can be used for automatic enhancement techniques such as the application of reflective textures to windows.*

## 1. Introduction

Automatic structure and motion recovery algorithms have matured rapidly over the past ten years, to the point that given an input sequence of images they can often produce a three dimensional model of the the scene [2, 11]. However there are many scenes for which they fail, typically due to ambiguity of the scene caused either by large homogeneous regions of texture, or repeated patterns in the image (which arise frequently in many man made structures). Occluded regions can also cause severe problems for many dense stereo algorithms [7]. A common approach to dense stereo is to combine a matching cost per pixel with a Markov random field prior and a model for occlusion[3, 4, 7, 8]. However this approach leads to very difficult optimization problems; algorithms which solve for MRF priors usually fail to adequately account for occlusion or enforce to constraints between adjacent epipolar lines, whereas a model based approach does both.

In this paper we propose a model based approach to structure from motion recovery in which priors on shape and texture are explicitly stated and used to overcome image ambiguities. In previous work [16] only the reprojection of the model into each image was used for its verification, whereas in this paper we propose that learnt statistics of the appearance of each model also be used to help determine the most appropriate model (and hence the shape of the scene). The combined use of correspondence and appearance data helps to more accurately identify which model is most appropriate.

In this paper the modelling of classical architecture, whose construction often conforms to a set of well defined rules, is used to illustrate these concepts. For example, this extract from Renaissance architect Andrea Palladio's Four Books of Architecture[10] dictates precisely the taper to be applied to columns: "if the column be fifteen foot high, the thickness at the bottom must be divided into six parts and a half, five and a half of which will be the thickness for the top". More general rules can also be applied, such as the fact that floors are horizontal, implying that windows generally occur in rows. Such rules provide strong prior information which can augment existing structure and motion techniques for architectural model acquisition. To encode this, a "Lego kit" of parameterised building blocks is used with prior distributions defined by the style of the building. The advantages of this model based approach are twofold. First, it enables accurate reconstruction of both the geometry and texture of parts of the scene which are occluded in some views. Second, while other model acquisition algorithms recover only the geometry and possibly texture of the scene, this algorithm also provides an interpretation of the scene. This leads to the generation of more photorealistic models

Proceedings of the Eighth IEEE International Conference on Computer Vision (ICCV'01)
0-7695-1143-0/01 $17.00 © 2001 IEEE

0-7695-1143-0/01 $10.00 (C) 2001 IEEE

e.g. appropriate transparency and reflectance properties can automatically be applied to regions of the model based on their interpretation. In addition, whereas there is no clear way of rendering a depth map without some postprocessing (simply joining adjacent pixels into triangles will result in a model containing over 250000 triangles for a 512x512 image), our model is easily manipulated and rendered as it is represented as a set of components which correspond to a natural decomposition of the scene into independent parts—the model shown in Figure 5 contains about 1000 triangles.

The highly constrained structure of architecture has attracted previous research in computer vision, most notably the Facade system [15] which creates convincing 3D models from a small number of simple polyhedral blocks and a sparse set of images. However, Facade is an interactive system which requires the manual placement of 3D blocks and their registration in each image, whereas this paper presents an automatic Facade-like system.

The rest of the paper is organised as follows: Section 3 defines an architectural model as a collection of wall planes containing parameterised shapes, and establishes a framework for optimising it. This framework combines prior probabilities for the texture (Section 4.1) and shape (Section 4.2) of the model with a likelihood measure based on the appearance of the model in each image (Section 4.3). In Section 5 it is shown how these cues are combined to estimate the MAP parameters for the model, while Section 6 describes how individual shapes in the model can be combined to form rows and columns. Finally, results are presented and discussed in Section 7.

## 2. Initialisation

An architectural scene is modelled as a set of base planes corresponding to walls or roofs, each of which may contain offset 3D shapes which model common architectural primitives (see Table 1, Figure 1). To estimate the base planes, corner and line features are detected and sequentially matched [2] across several views of the scene. The cameras are then self-calibrated[9] to obtain a sparse metric reconstruction. The base planes of the model are initialised by recursively segmenting planes from the reconstruction using a version of RANSAC constrained by architectural heuristics, e.g. that planes are likely to occur parallel or perpendicular to each other, and perpendicular to a common ground plane[6].

## 3. Problem formulation

After this initialisation step the base planes are considered fixed and our architectural model $\mathbf{M}$ contains parameters $\boldsymbol{\theta} = \{n, \boldsymbol{\theta}_L, \boldsymbol{\theta}_S, \boldsymbol{\theta}_T\}$, where $n$ is the number of primitives in the model, $\boldsymbol{\theta}_L$ is an identifier for the type of each primitive, $\boldsymbol{\theta}_S$ are structure parameters which define its shape and $\boldsymbol{\theta}_T$ are texture parameters describing its appearance. The texture parameters are intensity variables $i(\mathbf{x})$ (between 0 and 255) defined at each point $\mathbf{x}$ on a regular 2D grid covering the model surface. As the model is defined as a collection of primitives, it is useful to further decompose the parameters according to the primitives to which they belong: $\boldsymbol{\theta}_L = \cup_{i=1}^{n} \theta_L^i$, $\boldsymbol{\theta}_S = \cup_{i=1}^{n} \theta_S^i$ and $\boldsymbol{\theta}_T = \cup_{i=0}^{n} \theta_T^i$, where $\theta_T^0$ is the set of texture parameters belonging to the wall plane.

**Table 1.** Some primitives available for modelling classical architecture. Parameters in brackets are optional; a mechanism for deciding automatically whether they are used is given in Section 5.2. The parameters are defined as follows: $x$: $x$ position; $y$: $y$ position; $w$: width; $h$: height; $d$: depth; $a$: arch height; $b$: bevel (sloped edge); $dw$: taper of pillars, buttresses. The NULL model is simply a collection of sparse triangulated 3D points. $\mathcal{M}_0$ is reserved as the background model (generally a wall).

| $\theta_L^i$ | Description | $\theta_S^i$ |
|---|---|---|
| $\mathcal{M}_1$ | Window | $x, y, w, h, d, (b), (a)$ |
| $\mathcal{M}_2$ | Door | $x, y, w, h, d, (b), (a)$ |
| $\mathcal{M}_3$ | Pediment | $x, y, w, h, d$ |
| $\mathcal{M}_4$ | Pedestal | $x, y, w, h, d$ |
| $\mathcal{M}_5$ | Entablature | $x, y, w, h, d$ |
| $\mathcal{M}_6$ | Column | $x, y, w, h, d, dw$ |
| $\mathcal{M}_7$ | Buttress | $x, y, w, h, d, dw$ |
| $\mathcal{M}_8$ | Drainpipe | $x, w, h$ |
| $\mathcal{M}_9$ | NULL | $x_1 ... x_n, y_1 ... y_n, z_1 ... z_n$ |

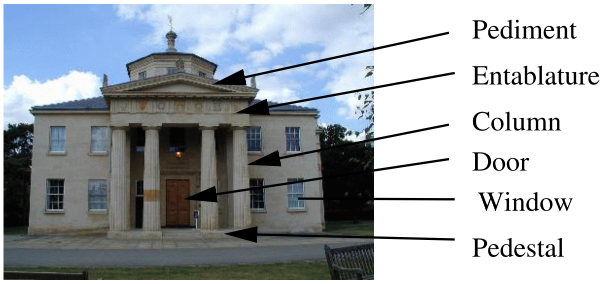


**Figure 1.** Example of some primitives used to construct classical architecture.

We require both the model $\hat{\mathbf{M}}$ which best models the scene and its optimal parameters $\hat{\boldsymbol{\theta}}$. Thus we want to maximise

$$\Pr\,(\mathbf{M}\boldsymbol{\theta}|\mathbf{DI})\ \alpha\ \Pr\,(\mathbf{D}|\mathbf{M}\boldsymbol{\theta}\mathbf{I})\,\Pr(\mathbf{M}\boldsymbol{\theta}\mathbf{I})$$

$$
\begin{aligned}
&= \Pr(\mathbf{D}|\mathbf{M}\boldsymbol{\theta}\mathbf{I})\Pr(\boldsymbol{\theta}|\mathbf{M}\mathbf{I})\Pr(\mathbf{M}\mathbf{I}) \\
&= \Pr(\mathbf{D}|\mathbf{M}\boldsymbol{\theta}_L\boldsymbol{\theta}_S\boldsymbol{\theta}_T\mathbf{I})\Pr(\boldsymbol{\theta}_L\boldsymbol{\theta}_S\boldsymbol{\theta}_T|\mathbf{M}\mathbf{I})\Pr(\mathbf{M}\mathbf{I}) \\
&= \Pr(\mathbf{D}|\mathbf{M}\boldsymbol{\theta}_L\boldsymbol{\theta}_S\boldsymbol{\theta}_T\mathbf{I})\Pr(\boldsymbol{\theta}_T|\boldsymbol{\theta}_L\mathbf{M}\mathbf{I}) \\
&\quad \Pr(\boldsymbol{\theta}_S|\boldsymbol{\theta}_L\mathbf{M}\mathbf{I})\Pr(\boldsymbol{\theta}_L|\mathbf{M}\mathbf{I}) \qquad (1)
\end{aligned}
$$

where $\mathbf{D}$ is the available data (the images), and $\mathbf{I}$ denotes prior information (the camera calibration and the estimated wall planes). Note that the probabilities of $\boldsymbol{\theta}_T$ and $\boldsymbol{\theta}_S$ are independent as the probability distribution for the texture of a primitive is unaffected by its shape, and vice versa. Each term in (1) has an intuitive interpretation:

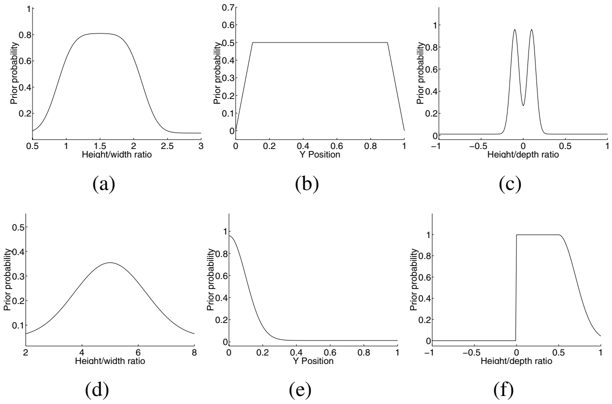
- $\Pr(\boldsymbol{\theta}_L|\mathbf{M}\mathbf{I})$ is the probability of each type of primitive. It may be used to specify the relative frequency with which primitive types occur, e.g. that windows are more common than doors, and is manually set to reflect the style of building being modelled (e.g. a Gothic building will have a high probability of buttresses where for a classical building columns are more likely).
- $\Pr(\boldsymbol{\theta}_S|\boldsymbol{\theta}_L\mathbf{M}\mathbf{I})$ is a prior on shape. This encodes prior knowledge of architectural style, for instance that windows in Gothic architecture are narrow and arched; it can also encode practical constraints, such as that doors generally appear at ground level. These priors are discussed in Section 4.1.
- $\Pr(\boldsymbol{\theta}_T|\boldsymbol{\theta}_L\mathbf{M}\mathbf{I})$ is probability of the texture parameters. This is evaluated using learnt models of appearance, such as the fact that windows are often dark with intersecting mullions (vertical bars) and transoms (horizontal bars), or that columns contain vertical fluting. This is described in more detail in Section 4.2.
- $\Pr(\mathbf{D}|\mathbf{M}\boldsymbol{\theta}_L\boldsymbol{\theta}_S\boldsymbol{\theta}_T\mathbf{I})$ is the likelihood of the images given a complete specification of the model. This is determined by the deviation of image intensities from the projection of the texture parameters, as described in Section 4.3.

The next section considers how these probabilities may be evaluated.

# 4. Evaluation of probabilities

## 4.1. Shape priors

The shape prior $\Pr(\boldsymbol{\theta}_S|\boldsymbol{\theta}_L\mathbf{M}\mathbf{I})$ is a product of distributions derived from both rules gleaned from architectural texts and practical considerations. Figure 2 gives some examples of these distributions for window and column primitives. The priors on height to width ratio are taken from [5], in which 1, 1.25, 1.5, 1.75 and 2.0 are given as common ratios, depending on the floor on which the window occurs; hence the prior for window height to width ratio in Figure 2(a) peaks at these values. These statements are confirmed by observation; for example the height to width ratio of the windows in Figure 1 is exactly 1.5. Figure 2(b) and (e) are examples of priors dictated by practical considerations: columns should appear at ground level or slightly elevated on a pedestal, whereas a window may occur at almost any elevation on a wall. Similar considerations also lead to strong prior distributions for other types of primitives.



(a) (b) (c)

(d) (e) (f)

**Figure 2.** Unnormalised prior distributions. (a) Height/width ratio of windows. (b) Elevation of windows. (c) Height/depth ratio of windows. (d) Height/width ratio of columns. (e) Elevation of columns. (f) Height/depth ratio of columns.

## 4.2. Texture priors from appearance models

The distribution Pr $(\boldsymbol{\theta}_T|\boldsymbol{\theta}_L\mathbf{M}\mathbf{I})$ is learnt from a training set of over 30 frontal images of classical architecture collected from both local sites and websites. In each training image, textures belonging to primitives and to the wall are manually marked and labelled. Textures from the same primitive type vary in appearance due to a number of factors, such as lighting and scale. To reduce the variation induced by these factors, a wavelet decomposition is applied to the strict interior of each texture. The 5/3 biorthogonal filter bank (defined in [13], also used in [12]) is chosen to effect the decomposition, as it has several desirable properties: (a) compact support (which allows accurate localisation of features); (b) linear phase (so that the orientation of filters at different scales remains constant); (c) its low pass filter has a zero-order vanishing moment, which cancels effects of global illumination changes. An example of two texture patches and their wavelet decompositions is given in Figure 3.

As in [12, 14], an appearance model is represented by

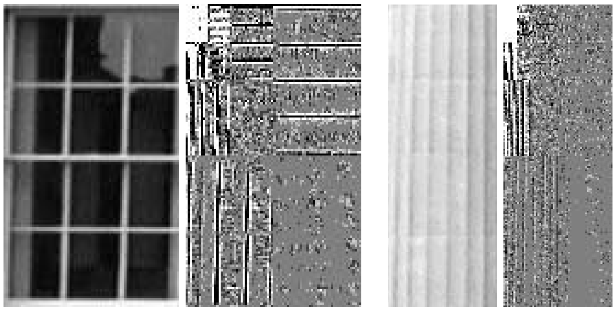


**Figure 3.** Window and column texture. Beside each texture is its corresponding (quantised) wavelet transform. Output is shown for 3 levels of the wavelet transform. At each level, the HL (horizontal high-pass, vertical low-pass) component is shown at the bottom left, the HH component at the bottom right, the LH component at the top right and the LL component at the top left. The HL component responds strongly to vertical edges, while the LH component detects horizontal edges. The LL component is a smoothed, subsampled version of the original texture.

a set of histograms. Each histogram counts the number of occurrences of every 3 × 3 pattern in a subband of the texture's wavelet decomposition. The decomposition output is quantised to limit the number of bins in each histogram. The prior probability of a patch of wavelet coefficients is then given by its frequency in the corresponding normalised histogram. Because texture is sampled regularly from the model surface and not the image, distortion due to the camera not being front on to the model surface is automatically nullified.

Some types of primitive have distinctive texture; for example in Edinburgh regions of wall were painted black with white vertical and horizontal strips, producing convincing illusions of windows[17]. However other types of primitive are not so easily distinguished by their texture, such as buttresses which are usually constructed from the same material as the wall they abut. Hence as in [14] texture at the edge of each highlighted primitive is recorded in a separate histogram, based on its smoothed intensity gradient perpendicular to the local boundary orientation. This allows even primitives without strong texture to be detected, as all primitives are likely to have edges at their boundary due to shadows or occlusion.

### 4.3. Evaluation of the likelihood

The likelihood $\Pr(\mathbf{D}|\mathbf{M}\boldsymbol{\theta}\mathbf{I})$ depends only on the texture parameters $\boldsymbol{\theta}_T$. It is evaluated by assuming that the projection of each texture parameter $i(\mathbf{x})$ into each image is corrupted by noise $\epsilon \in G(0, \sigma_\epsilon)$, thus $i(\mathbf{x}^j) = i(\mathbf{x}) + \epsilon$, where $i(\mathbf{x}^j)$ is the projection of $i(\mathbf{x})$ into image $j$. Assuming that the error $\epsilon$ at each pixel is independent, the likelihood over all points $\mathbf{x}$ is written:

$$\Pr(\mathbf{D}|\mathbf{M}\boldsymbol{\theta}\mathbf{I}) = \prod_j \frac{1}{\sqrt{2\pi}\sigma_\epsilon} \exp -\frac{1}{2}\left(\frac{i(\mathbf{x}^j) - i(\mathbf{x})}{\sigma_\epsilon}\right)^2 \tag{2}$$

## 5. Obtaining the MAP estimate

Having shown how each term in (1) is evaluated, we now formulate an algorithm, summarised in Algorithm 1, to search for a maximum a posteriori (MAP) estimate of both $\mathbf{M}$ and $\boldsymbol{\theta}$. Ideally the combined $\boldsymbol{\theta}_T$, $\boldsymbol{\theta}_S$ and $\boldsymbol{\theta}_L$ parameter space would be searched for the MAP parameters. However each primitive may contain thousands of texture parameters, so only $\boldsymbol{\theta}_S$ and $\boldsymbol{\theta}_L$ are searched. This is carried out in two steps: an initial search based on an approximate likelihood function (Section 5.1) locates likely values for a subset of the model parameters. These are then used to seed searches in the full parameter space using the complete likelihood function (Section 5.2).

### 5.1. Shape hypothesis from a single image

The search for MAP model parameters is initialised by sampling the shape parameters $\boldsymbol{\theta}_{S^1} = \{x, y, w, h\}$ at regular intervals, for each primitive type $\boldsymbol{\theta}_L$. The remaining shape parameters $d, a, b, dw$ are fixed at 0. Because estimating the texture parameters (Section 5.3) requires all of the shape parameters to be known, $\boldsymbol{\theta}_T$ is approximated as $\boldsymbol{\theta}_{T^1}$ which is the projection of each wall plane onto a single near frontal image $\mathbf{D}_1$. The single image likelihood $\Pr(\mathbf{D}_1|\boldsymbol{\theta}_{T^1}\boldsymbol{\theta}_{S^1}\boldsymbol{\theta}_L\mathbf{MI})$ is a good indicator of the full likelihood function $\Pr(\mathbf{D}|\boldsymbol{\theta}_T\boldsymbol{\theta}_S\boldsymbol{\theta}_L\mathbf{MI})$ because it is insensitive to errors in the fixed shape parameters $d, a, b, dw$. The projection of a primitive to a frontal view is affected only slightly by changes in its depth, and the $a, b, dw$ parameters are generally small compared to $x, y, w$ and $h$. $\Pr(\mathbf{D}_1|\boldsymbol{\theta}_{T^1}\boldsymbol{\theta}_{S^1}\boldsymbol{\theta}_L\mathbf{MI})$ is evaluated using the texture likelihood histograms defined in Section 4.2.

Likely values of $\theta^i_L$ and $\theta^i_{S^1}$ for primitive $i$ are recorded by sampling and ranking primitives of each type according to their single image likelihood ratio:

$$\mathcal{L}^i_S = \frac{\Pr(\mathbf{D}_1|\theta^i_{T^1}\theta^i_{S^1}\theta^i_L\mathbf{MI})}{\Pr(\mathbf{D}_1|\theta^i_{T^1}\theta^i_{S^1}\neg\theta^i_L\mathbf{MI})} \tag{3}$$

where $\neg\theta^i_L$ denotes every primitive type except $\theta_L$.

A list is maintained of the $N_R$ primitives with the highest ratio (3). $N_R$ is chosen to exceed the maximum number of primitives which are expected to appear on each wall plane. Some hypotheses are shown in Figure 4(a).

**Algorithm 1** Obtaining a MAP parameter estimate.

```
Stage 1: Plane initialisation (Section 2)
  Detect and sequentially match corner and line features[2], and self-calibrate cameras[9] to obtain a sparse reconstruction
  Recursively segment planes from the reconstruction using RANSAC, and heuristics applicable to architecture[1, 6].

for each segmented plane do
  Stage 2: In a single image: (Section 5.1)
  for each primitive type do
    Regularly sample parameters $x, y, w, h$
    Set remaining parameters $d, a, b, dw$ to 0.
    Rank model according to likelihood ratio $\mathcal{L}_S$ (3).
  end for

  Stage 3: In multiple images: (Section 5.2)
  for $N_R$ highest ranked primitives do
    Draw parameter $d^i$ from $\Pr(\theta_S^i|\theta_L^i\{x^i y^i w^i h^i\}\mathbf{MI})$, $i = 1..N_R$
    Search for maximum likelihood shape parameters $\{x^i, y^i, w^i, h^i, d^i\}$.
    Use model selection measure[16] to determine value of remaining primitives $\{a^i, b^i, dw^i\}$
    Threshold the primitive based on likelihood ratio $\mathcal{L}_M$ (4).
  end for

  Detect and model dependency between primitives occurring in rows or columns. (Section 6)
end for
```

## 5.2. Shape refinement from multiple images

In multiple views, maximum likelihood shape parameters $\boldsymbol{\theta}_S^{ML}$ are found for each primitive proposed from a single image. For each primitive $i$, the depth parameter $d^i$ is drawn from its prior distribution $\Pr(\theta_S^i|\theta_L^i\{x^i, y^i, w^i, h^i\}\mathbf{MI})$. A direct search algorithm is then used to find local maximum likelihood values for $\{x^i, y^i, w^i, h^i, d^i\}$ from this seed point. The parameter $d^i$ is drawn multiple times for each hypothesis, and multiple searches are used, to find the global maximum.

Having maximised the likelihood of each primitive, a model selection criterion is used to decide whether the optional parameters $a, b, dw$ should be included. [16] discusses how AIC, BIC and Occam factors can be used for this. The idea is that extra parameters are penalised, and are included in the model only if the resulting improvement the maximum likelihood estimate outweighs this penalty. For example using the AIC measure the term

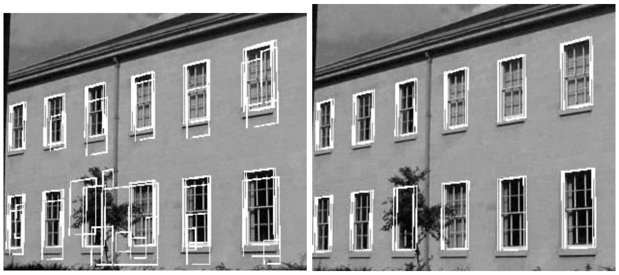


**Figure 4.** (a) Collection of 50 most likely primitives found by single image proposal process. (b) After merging overlapping hypotheses and thresholding on multiple image likelihood ratio.

$(\log \Pr(D|\mathbf{M}\boldsymbol{\theta}^{ML}\mathbf{I}) - 2N)$ is optimised rather than simply the likelihood, where $N$ is the number of parameters in the model.

## 5.3. Estimation of the texture parameters

Having estimated both $\boldsymbol{\theta}_L$ and $\boldsymbol{\theta}_S$, only the texture parameters $\boldsymbol{\theta}_T = \{i(\mathbf{x})\}$ remain to be found. It is assumed that each texture parameter is observed with noise $i(\mathbf{x}) + \epsilon$, where $\epsilon$ has a Gaussian distribution mean zero and standard deviation $\sigma_\epsilon$. Each parameter $i(\mathbf{x})$ can then be found such that over $m$ views it minimizes the sum of squares $\sum_{j=1}^{j=m}\left(i(\mathbf{x}^j) - i(\mathbf{x})\right)^2$ where $i(\mathbf{x}^j)$ is the intensity at $\mathbf{x}^j$, and $\mathbf{x}^j$ is the projection of $\mathbf{x}$ into the $j$th image.

Finally, each hypothesised primitive is thresholded based on its multiple image likelihood ratio:

$$\mathcal{L}_M^i = \frac{\Pr(\mathbf{D}|\theta^{ML,i}\mathbf{MI})}{\Pr(\mathbf{D}|\theta^i\mathbf{M}_0\mathbf{I})} \qquad (4)$$

where $\mathbf{M}_0$ is the model containing no primitives. This eliminates spurious hypotheses found in a single image, as shown in Figure 4.

# 6. Detecting rows and columns

So far each primitive has been detected, classified and optimised individually. However identical primitives often occur in rows or columns, allowing for a more compact and robust parametrization of the model as several primitives can be represented by a single set of parameters. Primitives are grouped into rows and columns by taking the $x$ and $y$ parameter of each shape in turn, and counting the number of remaining shapes of the same type with a similar $x$ or $y$ position. Rows and columns are ranked according to the number of primitives they contain and the largest group is retained. This process is repeated for the remaining primitives until all rows and columns have been tested. After primitives have been grouped, they are assigned global parameters which are then optimised. The added robustness this gives to parameter estimates is shown in Figure 7, where both geometry and texture information is propagated from the nearest window in the model to improve the appearance of the distant windows which are partially occluded in all views. In future we hope to incorporate this and more sophisticated groupings into a probabilistic framework by the use of hierarchical models for more complex structures such as an entire classical entrance or Gothic facade constructed from individual primitives.

## 7. Results

Figure 5 shows some renderings of a 3D architectural model, recovered from 5 high resolution (1600x1200) images. This model has some nice properties:

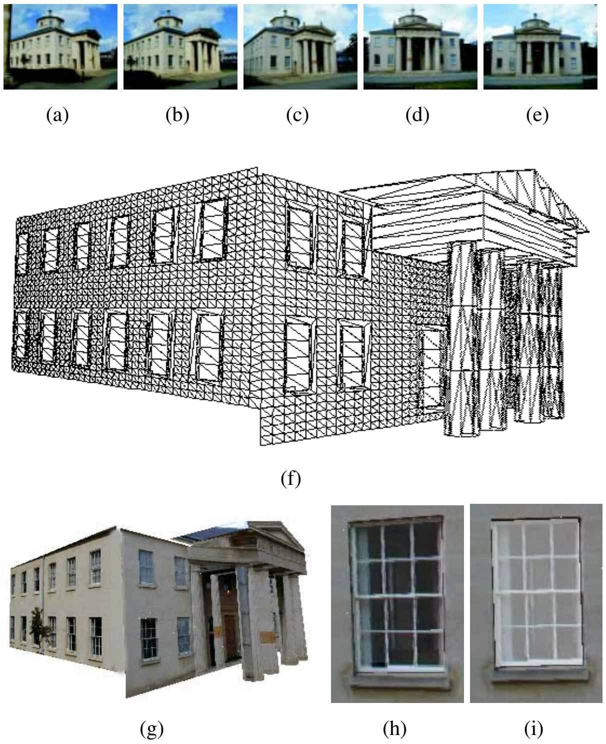
- The wireframe is constructed from a collection of simple primitives which have been assigned an interpretation, rather than as an unstructured cloud of 3D points. This makes it very simple to render the model, and to automatically add enhancements to it. For example each window has been made reflective and partially transparent, so that it exhibits specular reflection as the viewer moves past it, as can be seen in Figure 5(h)-(i).

- The columns at the entranceway are correctly reconstructed despite only their front face being visible in the images. This is only possible due to the use of strong prior models for shape.

- In addition to the two base planes, there are 25 extra primitives, gathered into 5 rows, containing a total of 70 parameters, i.e. fewer parameters than a set of 24 3D points in general position. Due to its compactness the model is also accurate—the wall plane is represented as a plane rather than a collection of near coplanar points, for instance.

Figure 6 shows another model of classical architecture. Again the recovered structure is accurate and compactly represented, and each window has been identified. The arch feature was also identified as a window because the set of primitives includes only arched windows and doors, and its elevation precludes it from being labelled as a door.

A model of Gothic architecture is shown in Figure 7. Much of the side wall of the chapel is obscured by buttresses and vegetation in most images, so that there is no texture available for parts of the distant windows. This is remedied by pasting texture from the window with the highest likelihood ratio onto windows which are detected as belonging to the same row of primitives, thus improving the appearance of the model.


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

(f)

(g) (h) (i)

**Figure 5.** (a)-(e): Original views of the Library model. (f) Wireframe rendering. Each window has been accurately recovered, as well as the components of the entranceway. The left wall plane is truncated where it is obscured by vegetation which is not coplanar with the wall. (g) With texture applied. (h)-(i) Reflective texturing is automatically applied to windows. Hence the appearance of this window changes as the viewer moves past it.

## 8. Conclusion

This paper has presented an automatic system for constructing and interpreting photorealistic models of architecture from a "Lego kit" set of building blocks. A probabilistic framework has been developed which incorporates priors on shape, derived from architectural principles, and priors on texture, based on learnt appearance models. An algorithm has been proposed for finding the MAP estimate of both the model and its parameters based on this prior information and parallax information from image data. Models are recovered as a set of labelled parts rather than a dense collection of 3D points, presenting many possibilities for manipulating and rendering architectural models.

Ongoing work includes the extension of this formulation, possibly to include more sophisticated priors Pr $(\boldsymbol{\theta}_L|\mathbf{MI})$ differentiating architectural styles. Further model enhancement techniques are also being considered, such as the gen-

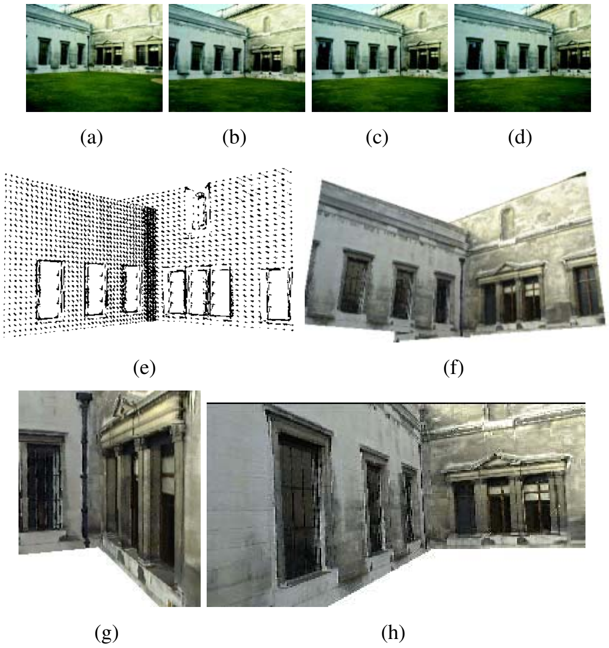


**Figure 6.** (a)-(d): Original views. (e) Wireframe model. (f) Textured model. (g)-(h) Details showing recovered windows.

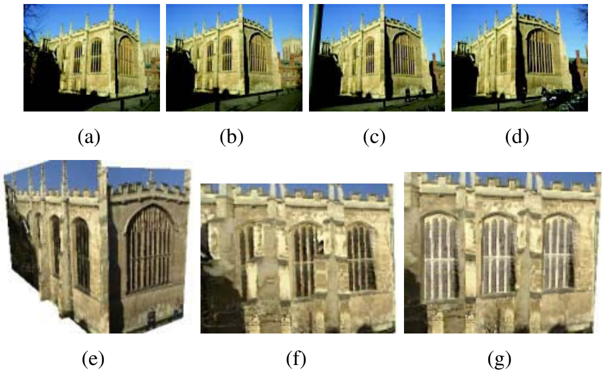


**Figure 7.** (a)-(d): Original views. (e) Textured model. (f) Original texture pasted from one view. (g) Texture of occluded windows is enhanced by pasting texture from unoccluded window.

eration of synthetic textures for each primitive type. This would greatly improve the appearance of models which are currently textured from photographs and hence are subject to lighting changes, shadows and occlusion.

## References

[1] C. Baillard and A. Zisserman. Automatic reconstruction of piecewise planar models from multiple views. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 559–565, June 1999.

[2] P. Beardsley, P. Torr, and A. Zisserman. 3d model acquisition from extended image sequences. In *European Conference on Computer Vision*, pages II:683–695, 1996.

[3] P. Belhumeur. A bayesian-approach to binocular stereopsis. *International Journal of Computer Vision*, 19(3):237–260, 1996.

[4] I. Cox, S. Hingorani, S. Rao, and B. Maggs. A maximum-likelihood stereo algorithm. *CVIU*, 63(3):542–567, 1996.

[5] D. Cruickshank and P. Wyld. *London: the Art of Georgian Building*. Out of print., 1975.

[6] A. Dick, P. Torr, and R. Cipolla. Automatic 3d modelling of architecture. In *Proc. 11th British Machine Vision Conference (BMVC'00)*, pages 372–381, Bristol, 2000.

[7] H. Ishikawa and D. Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *ECCV98*, pages 232–249, 1998.

[8] R. Koch, M. Pollefeys, and L. Van Gool. Multi viewpoint stereo from uncalibrated video sequences. In *European Conference on Computer Vision*, pages 55–71, 1998.

[9] P. Mendonca and R. Cipolla. A simple technique for self-calibration. In *CVPR99*, pages I:500–505, 1999.

[10] A. Palladio. *The Four Books on Architecture*. MIT Press, 1997. Translated by R. Tavernor and R. Schofield.

[11] M. Pollefeys, R. Koch, M. Vergauwen, and L. van Gool. Metric 3d surface reconstruction from uncalibrated image sequences. In *SMILE Workshop*, pages 139–155, 1998.

[12] H. Schneiderman and T. Kanade. A statistical method for 3d object detection applied to faces and cars. In *CVPR00*, pages I:746–751, 2000.

[13] G. Strang and T. Nguyen. *Wavelets and Filter Banks*. Wellesley-Cambridge Press, Wellesley, 1996.

[14] J. Sullivan, A. Blake, M. Isard, and J. MacCormick. Object localization by bayesian correlation. In *ICCV99*, pages 1068–1075, 1999.

[15] C. Taylor, P. Debevec, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. *SIGGraph*, pages 11–20, 1996.

[16] P. Torr, A. Dick, and R. Cipolla. Layer extraction with a bayesian model of shapes. In *European Conference on Computer Vision*, pages II:273–289, 2000.

[17] P. Wilkinson. *The Shock of the Old*. Channel 4 Books, 2000.