Advancing Structured Prediction for 3D Polygonal Model Reconstruction from Single and Multiple Scene Images

Debjyoti Chattopadhyay[1], Isha Sahni[2] and Arpan Dutta Chowdhury[3]

debjyotisonuabhi@gmail.com[1] ishasahni2000@gmail.com[2] arpandc14@gmail.com[3]
Gupshup , Goregaon, Mumbai, Maharashtra[1] , IBM , Bhartiya City Bengaluru[2] , Tata Unistore, Mumbai, Maharashtra[3]

Abstract. We introduce a novel approach for structured prediction aimed at reconstructing 3D polygonal models from both single and multiple images capturing a scene. Building upon recent advancements in single-view reconstruction, we embrace the indoor Manhattan hypothesis category – an intricate set of potential outputs characterised by complex internal constraints – integrated into a structured prediction framework. Our methodology is adaptable for learning in both single-view and multiview scenarios. It is demonstrated that this chosen hypothesis category enables the optimization of diverse high-level loss functions, including metrics such as the relative depth error. Our achieved outcomes surpass the current state-of-the-art, showcasing an enhancement of over 50% in a specific metric.

Keywords: Computer Vision,3D-model reconstruction,multiple–view reconstruction

1 Introduction

Reconstruction of 3D models from images is a central problem in computer vision. There is a deep literature concerning reconstruction from multiple views of a scene, with particular focus on the geometry of multiple cameras. More recently the recovery of 3D information from a single image has received attention [1,2]. In this context the geometric constraints leveraged by multiple–view reconstructors are unavailable, so inference must rely on photometric cues. To capture the complex, high–dimensional patterns involved in this challenging inference task, practitioners have leveraged a range of machine learning techniques.

Within a single–view reconstruction, few authors have cast learning as a single optimisation with respect to a clearly defined loss function [1–4], while most approaches to multiple–view reconstruction do not consider learning from training data at all. In contrast, this paper casts reconstruction fundamentally as a learning problem, with the goal being to learn a prediction function *of mapping* observed features to 3D reconstructions.

JRI 2023, December 18-20, Ouagadougou, Burkina Faso

DOI 10.4108/eai.18-12-2023.2348109

Building on recent advances in single–view reconstruction, we adopt as our hypothesis class the set of indoor Manhattan models [3,5,4], under which scenes are approximated by a floor and ceiling plane together with a sequence of vertical walls. This representation brings a variety of attractive features such as a simple parameterization, efficient and exact inference, decomposability of loss functions, and a balance between expressiveness and robustness.

For learning we use the tools of structured prediction, in particular the structural SVM [6]. The use of these tools is part of a long trend towards statistically rigorous, well–understood convex optimization techniques in computer vision. Recent successful applications include detection [7], segmentation [8], and scene classification [9]. In the domain of reconstruction, structured prediction ideas have been applied to several simple model classes such as stereo disparities [10] and cuboids [11].

The application of structured prediction to the indoor Manhattan class of models constitutes one of the most complex output spaces yet considered within this framework. The indoor Manhattan model enforces hard geometric constraints that lack simple expressions in terms of image coordinates. These constraints are context–dependent, being tied to quantities such as camera rotation and the location of vanishing points. We have learnt several valuable lessons of general relevance from this complex prediction task, to which we dedicate the final section of this paper.

The contributions of this paper are thus (i) a unified learning framework for single– and multiple–view reconstruction, using the indoor Manhattan model and the tools of structured prediction; (ii) the reduction of two image–level loss functions to a form amenable to efficient optimization; (iii) an efficient separation procedure for identify the "most–violated constraint" during learning, (iv) an empirical demonstration of structured prediction in perhaps the most complex output space yet considered within this framework; and (v) a series of practical observations concerning the application of structured prediction techniques.

In the remainder of this paper we present background material (Section 2), followed by the indoor Manhattan model itself (Section 3), and our learning framework (Section 4). We then present results for multiple–view reconstruction (Section 5) followed by single–view reconstruction (Section 6), then we round off with practical lessons learnt (Section 7) and concluding remarks (Section 8).

2 Background

This paper deals with several major research areas (multiple–view reconstruction, single–view reconstruction, and structured prediction), so here we present only key contributions and those results of particular relevance to our own work.

Multiple view reconstruction has a long history in the literature, beginning at least as early as the seminal work of Marr *et al.* [12]. Low–level approaches estimate the depth of each pixel by solving for stereo disparity (for a survey see [13]), while high–level approaches attempt to recover polyhedral models of various kinds [14,15]. The present work follows the spirit of the latter approach.

Coughlan and Yuille [16] first introduced the Manhattan world assumption, in which reconstructed surfaces are restricted to three mutually–orthogonal orientations. Furukawa *et al.* [17] applied this idea within a multiple view stereo context. Lee *et al.* [3] proposed the indoor Manhattan assumption, which places further restrictions on reconstructed models.

Single–view reconstruction from line drawings also has a long history in the literature ([18], for example). Reconstructing single real–world images was re–introduced to the community by Hoiem *et al.* [1], who employed decision trees to classify image segments into orientation classes in a multiple segmentation approach. Saxena *et al.* [2] employed energy minimisation to recover pixel–wise depths. Barinova *et al.* [19] reconstructed piece–wise planar outdoor scenes using an EM algorithm. Hedau *et al.* [11] modeled indoor scenes as the interior of a cuboid, and cast learning within a structured prediction framework. We also use structured prediction, though our hypothesis class is far more expressive.

Lee *et al.* [3] were the first to reconstruct indoor Manhattan scenes. They devised a combinatorial search over line segments using a branch–and–bound algorithm. Flint *et al.* [5] refined this to an exact dynamic programming solution, and in later work [4] extended the indoor Manhattan assumption to the multiple view domain. The authors of [4] did describe a learning algorithm based on bootstrapping, though its statistical consistency was given little attention. The present work relies extensively on this thread of research, though in contrast to past work our focus is entirely on learning in a statistically rigorous framework. We provide comparisons with these approaches.

Extending binary and multi–label classification to general output spaces is a major research programme within the machine learning community; an excellent introduction is given by Bakir *et al.* [20]. Our approach uses the structured support vector machine, first proposed by Tsochantaridis *et al.* [6] and improved upon by Joachims *et al.* [21].

In this work, we present a novel unified learning framework designed for the reconstruction of polygonal models from both single and multiple views of a scene. Our approach introduces a distinctive loss function within a structured prediction framework, emphasizing a single optimization process with clear and well-defined objectives. Notably, we focus on the indoor Manhattan class of models, leveraging its parametrization and inference algorithm to enhance re- construction accuracy. The versatility of our framework is demonstrated through its application to both multiple-view and single-view reconstructions, showcasing its adaptability to various scenarios. Experimental results indicate a significant improvement over existing methods, emphasizing the effectiveness and unique- ness of our proposed approach in the domain of computer vision and scene understanding.

3 Model

In this section we describe three components of the model that we are trying to learn (and, at test time, infer): a hypothesis class, a feature space, and a loss function. In our setup these are, respectively, the class of indoor Manhattan models, a log–linear

Bayesian likelihood, and either the relative depth error or a labelling error (we describe both).

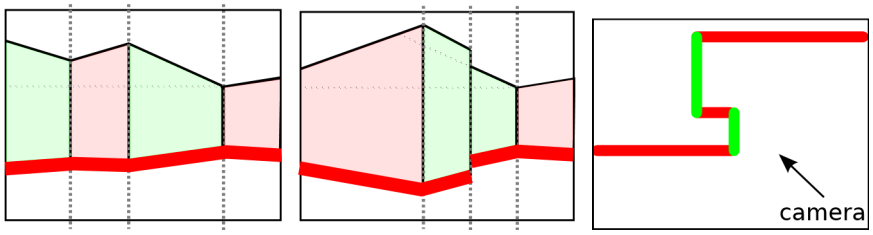

Fig.1: (a-b) Examples of indoor Manhattan environments. The red line illustrates the seam representation. (c) An indoor Manhattan environment viewed from above.

3.1 Hypothesis Class

This paper is concerned with the hypothesis class consisting of indoor Manhattan reconstructions, which are 3D polygonal models characterised by infinite floor and ceiling planes with vertical walls extending between them (as originally proposed by Lee *et al.* [3]). Indoor Manhattan environments are a sub–class of general Manhattan environments; examples are shown in Figure 1. This is an attractive hypothesis class because

1. it captures many regularities within man–made environments;
2. the geometric primitives (floor/wall/ceiling) are immediately useful for semantic–level scene understanding;
3. it is expressive enough to represent approximately or exactly a surprisingly wide variety of environments;
4. there is a simple and convenient parameterization;
5. an efficient inference algorithm exists [5].

An indoor Manhattan model is much like an architectural floor–plan, and can be specified as a set of 2D line segments representing walls together with the position of the floor and ceiling plane (Figure 1). In this paper we adopt an image–domain parameterization due to its convenience for inference and learning. Following Flint *et al.* [4] we represent hypotheses as a path running from the left edge to the right edge of the image, which we will refer to as a *seam*. A seam S consists of a sequence of pairs of scalars,

$$S=\{s_i,o_i\}_{i=1}^{W} . \quad (1)$$

where s_i is the y–coordinate at which the path intersects image column i, o_i is the orientation of the wall in that column, and W is the image width. Remarkably, this simple parameterization specifies a unique metric 3D model up to scale [4].

While we do not here have space for a full discussion of the geometry of indoor Manhattan models, there are two properties of the seam representation that will be

relevant in the following sections. Firstly, images of indoor Manhattan environments can be rectified so that vertical lines in the world project to vertical lines in the image [5]. Secondly, the mapping from seam to 3D reconstruction decomposes in the following manner. We said earlier that a seam S specifies a unique 3D reconstruction. Let $a(x,y;S)$ be the orientation of the 3D surface projecting to pixel (x,y). Flint *et al.* [5] showed that a is functionally dependent *only* on the pair (s_x,o_x) for column x. That is, we may write

$$a(x,y;S) = \tilde{a}(x,y;s_x,o_x) \; . \tag{2}$$

Similarly, letting $d(x,y,;S)$ be the distance from the camera to the surface projecting to (x,y) we may write

$$d(x,y;S) = \tilde{d}(x,y;s_x,o_x) \; . \tag{3}$$

3.2 Feature Space

We adopt the probabilistic model formulated by Flint *et al.* [4], which relates indoor Manhattan reconstructions to single–view image features, multiple–view photo–consistency terms, and a reconstructed point cloud. From our perspective these are simply features and any subset may be omitted if inappropriate for a given application. Under this model the prior on reconstructions is

$$\log P(S) = -n \cdot \lambda + O(1) \; . \tag{4}$$

where n is a vector containing the number of walls in S of various categories (concave, convex, occluding) and λ is a hyper–parameter. We may interpret (4) as, roughly, that reconstructions with more walls are less likely.

Flint *et al.* [4] showed that for a variety of features θ (including single–view and multiple–view) there are reasonable choices of likelihood that can be written

$$\log P(\theta \mid S) = \sum_{x=1}^{W} \pi_{\Theta}(x,s_x) \; . \tag{5}$$

where π_θ is a real matrix computed deterministically from θ and $\pi_\theta(i,j)$ is the element at row i, column j. It is easy to show that for each sensor model described in [4] the log–likelihood is linear in the hyper–parameters κ, *i.e.*

$$\log P(\theta \mid S) = \sum_{x=1}^{w} \kappa_{\Theta} \, v_{\Theta}(x,s_x) \; . \tag{6}$$

Assuming conditional independence between features, the posterior is

$$P(S \mid \Theta) \propto P(S) \prod_{i=1}^{n} P(\theta_i \mid S) \; . \tag{7}$$

$$\log P(S \mid \Theta) = -n \cdot \lambda + \sum_{i=1}^{n} \sum_{x=1}^{w} v_i(x,s_x) + O(1) \; . \tag{8}$$

Table.1: The composition of our single– and multiple–view feature space. We omit colour and Gabor features from the multiple–view feature space for training efficiency.'

Feature	Dimensionality	Multi–view ?	Single–view?	Reference
Stereo photo–consistency	4	yes	no	Flint *et al.* [4]
Point cloud	2	yes	no	Flint *et al.* [4]
Line sweeps	1	yes	yes	Lee *et al.* [3]
RGB+HSV	6	no	yes	
Gabor responses[4]	12	no	yes	

So far, this model closely follows that described in [4]; our contribution is to place this model into a statistically rigorous learning framework. To do this we need to rewrite the above in terms of a joint feature function $\Psi(\Theta,S)$ and a parameter vector w. The decomposability of indoor Manhattan models into payoff matrices permits precisely such a formulation. Defining

$$\Psi(\Theta,S)=\begin{bmatrix} -\boldsymbol{n} \\ \sum_{x=1}^{W}\nu_1(x,s_x) \\ \vdots \\ \sum_{x=1}^{W}\nu_n(x,s_x) \end{bmatrix} \qquad \boldsymbol{w}=\begin{bmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\kappa}_1 \\ \vdots \\ \boldsymbol{\kappa}_n \end{bmatrix} \tag{9}$$

we see that (8) can be written as

$$logP(S \vee \Theta) = w, \Psi(\Theta, S) + O(1)\,. \tag{10}$$

where we have adopted Hilbert space notation with $\langle\cdot,\cdot\rangle$ denoting an inner product. Since w contains all free parameters in the model, the goal of learning will be to optimise w with respect to a training set $\{(\Theta_i,S_i)\}$.

Features The precise make–up of the feature space depends on the available sensor modalities. We define separate feature spaces for the single– and multiple– view contexts; these are summarised in Table 2.

3.3 Loss Functions

Next we define a loss function $\Delta(S,\hat{S})$ measuring the cost of predicting some reconstruction $\hat{S}$ when in fact the true reconstruction is S. In the context of learning one often faces a trade–off between choosing a loss that leads to tractable optimization, and choosing a loss that measures the quantity that one "really" cares about. For example, Hoiem *et al.* [1] learn a per–segment orientation classifier, then pass this as input to a separate 3D reconstruction system [22]. However, what one "really" cares about is some loss defined on the output of the entire system rather than the output of individual components, since some segment– level mistakes are insignificant to the overall reconstruction quality, while others are catastrophic. This is not a criticism of the authors' choice, but an illustration of the trade–off faced when choosing a loss. In this paper we show how to learn efficiently with respect to a loss defined on the final reconstruction.

The relative depth error has been the gold standard within the reconstruction community for more than a decade [23], and measures the average deviation between reconstructed and ground truth depths. In our notation,

$$\Delta_{\text{depth}}(S,\hat{S}) = 1/N \, \Sigma_p \, \{ \, |d(p; \hat{S}) - d(p; S)| \, / \, d(p; S) \, \} \, , \qquad (11)$$

[4] 4 orientations, 3 scales

where N is the number of pixels. Another reasonable choice is the labelling error, used widely within the semantic segmentation literature,

$$\Delta_{\text{labelling}}(S, \hat{S}) = 1/N \, \textstyle\sum_p \{ \, a(p; \hat{S}) \neq a(p; S) \, \} \, , \qquad (12)$$

where $[p]$ is 1 if p is true and 0 otherwise. An attractive characteristic of the indoor Manhattan class is that *both of these losses can be optimised exactly*. The algorithmic details are left to Section 4; the key result we establish here is that

Δ_{depth} and $\Delta_{\text{labelling}}$ can be written in a form resembling the payoff formulation (5) for the feature likelihoods.

First we invoke the independence established in (3):

$$\Delta_{\text{depth}}(\hat{S}, S)=1/N \, \textstyle\sum_{x=1}^{w} \sum_{y=1}^{H} \{| \, \eth(x, y; \hat{s}_x) - d(x, y; S)|\}/\{d(x, y; S)\} \, . \qquad (13)$$

Defining a real matrix δ_S,

$$\delta_s(x, j) = \sum_{y=1}^{H} \{| \eth(x, y; \hat{s}_x) - d(x, y; S)|\}/\{d(x, y; S)\} . \quad (14)$$

we see that we can write Δ_{depth} in the form

$$\Delta_{depth}(\hat{S}, S)=1/N \sum_{x=1}^{W} \delta_s(x, \hat{s}_x) . \quad (15)$$

There is a similar form for $\Delta_{labelling}$, which we omit here due to space constraints.

Choosing a Loss Function Neither of the above losses is unequivocally the "correct" loss; the choice will depend on the application. One might expect a strong correlation between the losses, and indeed one can show analytically that

$$\Delta_{depth}(S,\hat{S}) = 0 \Longleftrightarrow \Delta_{labelling}(S,\hat{S}) = 0 . \quad (16)$$

However, in our experiments we found only a weak correlation between these losses away from the origin. For example, the scatter plot shown in Figure 2 shows a significant number of outliers that score very well on Δ_{depth} but poorly on $\Delta_{labelling}$, and vice versa.

4 Learning

We turn now to the problem of learning within the model described above. Our learning task is to identify a prediction function f mapping observed features Θ to reconstructions S. We seek the loss minimizer

$$f^* = \operatorname*{argmin}_f \mathbb{E}\Big[\Delta\big(f(\Theta), S\big)\Big] , \quad (17)$$

which we approximate in the framework of empirical risk minimization as

$$f^* = \operatorname{argmin}_{(f)} \sum_k \Delta((f(\Theta_k), S_k) , \quad (18)$$

where k indexes a training set. To perform this optimization we turn to the tools of structured prediction [20], and in particular the structured SVM [6]. First we need to define f. In this paper we consider predictors of the form

$$f_w(\Theta) = \operatorname{argmax}_{(S)} \langle w , \Psi(\Theta, S) \rangle . \quad (19)$$

Comparing (10) we see that each predictor of the form (19) is simply implementing MAP inference under some set of hyper–parameters *w*. We now turn to the optimization problem itself. Following the standard approach [20] we cast the learning problem as a constrained optimisation problem,

$$\min_{(w,\xi)} \; \tfrac{1}{2} \|w\|^2 + C \sum_{k=1}^{n} \xi \tag{20}$$

s.t. $\forall k, S \neq Sk : \; \langle w , \Psi(\Theta, S) \rangle \; - w, \Psi (\Theta_k , S) \geq \Delta(S, S_k) - \xi_k .$

Tsochantaridis *et al.* [6] described an algorithm for solving this minimisation that is now used extensively within machine learning and computer vision. To apply this algorithm here we must solve two inference problems:

1. *Prediction.* This is the maximisation described in (19).
2. *Separation.* The algorithm described in [6] requires a user–supplied procedure to find the "most–violated constraint" at each iteration. That is,

$$\operatorname{argmax}_{(S)} \; w, \Psi (\Theta_k , S) + \Delta(S, S_k) . \tag{21}$$

Our solutions to both of the above build on the algorithm presented by Flint *et al.* [5,4], which is a dynamic programming solution to problems of the form

$$\operatorname{argmax}_{(S)} \; \sum_x \pi(x, s_x) - \sum_j \gamma(j; S) . \tag{22}$$

4.1 Inference (Prediction)

We showed in Section 3 that (22) can be written in the form (19), so the prediction problem is a straightforward application of [5]. This is as expected since, as we have already remarked, (19) is equivalent to MAP inference on indoor Manhattan reconstructions, which was precisely the subject of [5].

4.2 Loss–Augmented Inference (Separation)

It turns out that the separation problem can also be solved using the dynamic programming algorithm mentioned above, as the following proposition shows.

Proposition 1. *Let* (Θ_k, S_k) *be a training instance with payoff matrices* $\{\pi_i\}$ *as defined in* (5). *Let*

$$\pi_{aug} = \delta_{S_k} + \sum_{xi} \pi_i . \tag{23}$$

Then the solution to (22) *with* $\pi = \pi_{aug}$ *is identical to the solution to* (21).

Proof. Direct equivalence of the expressions to be maximised. First substitute (10) and (15) into (22):

$$\log P(S \mid \Theta) + \sum_{x=1}^{w} \delta_{Sk}(x, s_x) \; . \tag{24}$$

Further substituting (5) and defining γ as in [4] gives

$$\sum_{i} \sum_{x=1}^{w} \pi i \left(x, sx\right) - \sum_{j} \gamma\left(j; S\right) + \sum_{x=1}^{w} \delta_{Sk} \left(x, s_x\right) \; . \tag{25}$$

Finally we see that substituting (23) gives

$$\sum_{x=1}^{w} \pi aug \left(x, sx\right) - \sum_{j} \gamma(j; S) \; . \tag{26}$$

5 Multiple View Results

We evaluated our approach on the data–set proposed in [4], which consists of 18 sequences of six environments averaging 59 seconds in duration. We sampled key–frames at regular intervals. Each “instance” in our training and hold–out sets consists of one base frame together with four auxiliary frames.

We compared it with the bootstrapping approach described in [4]. Our metrics differ from theirs in two ways. Firstly, they compute relative depth error using the maximum of the ground truth and estimated depths in the denominator, whereas we always use the ground truth in the denominator. These metrics are separated by at most a monotonic transform but the latter is more convenient

Table.2: Multiple–view reconstruction performance on held–out data, compared with Flint *et al.* [4]. For unavoidable reasons we use slightly different metrics so our figures differ from those published in [4]. See main text for explanation.

	Depth Error (%)		Labelling Error (%)	
Sequence	This Paper[2]	Flint *et al.*	This Paper[3]	Flint *et al.*
ground	4.9	66.6	2.9	10.4
foyer1	6.1	6.6	3.1	3.1
foyer2	4.3	5.4	3.7	4.0
corridor	14.6	52.9	9.5	19.2
mcr	34.0	67.6	15.	16.2
kitchen	16.8	23.6	5.2	6.1
Average	**13.4**	37.1	**6.7**	9.8

Table.3:Single-view reconstruction performance on held-out data, compared with Flint et al.[5]

	Depth Error (%)		Labelling Error (%)	
Sequence	This Paper[2]	Flint *et al.*	This Paper[3]	Flint *et al.*
ground	17.3	24.5	7.8	12.2
foyer1	25.1	31.0	15.1	22.2
foyer2	29.1	30.1	15.9	18.6
corridor	31.7	33.6	19.3	24.8
mcr	70.1	45.9	26.7	20.8
kitchen	25.1	26.2	7.7	11.9
Average	33.1	**31.9**	**15.4**	18.4

represent in our framework. Secondly, when we compute labelling error we differentiate vertical and horizontal surfaces only, whereas they also differentiate the two vertical orientations. The latter approach makes a side–by–side comparison difficult because the two vertical orientations are symmetric and their labels can always be interchanged.

The performance of these two algorithms are summarised in Table 2. Our approach significantly out–performs the bootstrapping algorithm. Anecdotally we noticed that much of the improvement resulted from a reduction in catastrophic failures. This makes sense because we would expect the learning algorithm to concentrate on

reducing those mistakes that result in the largest loss. Some example predictions are shown in Figure 4; many more are included in additional material.

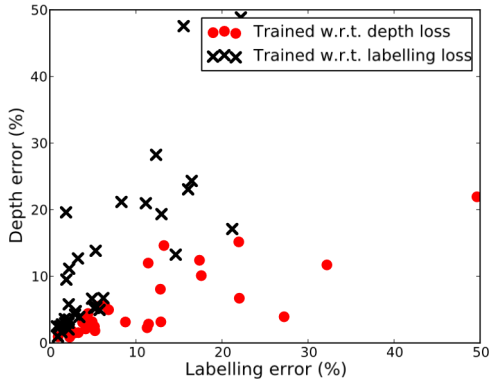

Fig.2: The effect of the loss function on training. We train two predictors, one with respect to Δ_{depth} and one with respect to $\Delta_{labelling}$, then evaluate both on all held–out instances. Each data point shows the error obtained by one predictor on held–out instances. The differing distribution of errors shows that the two predictors trade off errors as expected.

6 Single View Results

We evaluated our system for single–view reconstruction using the same data–set described in the previous section. We used the single–view features summarised in Figure 2. We compared our approach to the single–view approach of Flint *et al.* [5], which uses the same dynamic programming algorithm that we rely upon, but uses hand–tuned features.

Performance for each algorithm is summarised in Figure 3. When measured by labelling error, our approach out–performs the hand–tuned weights, but on the depth error metric our approach is inferior. While investigating this result we

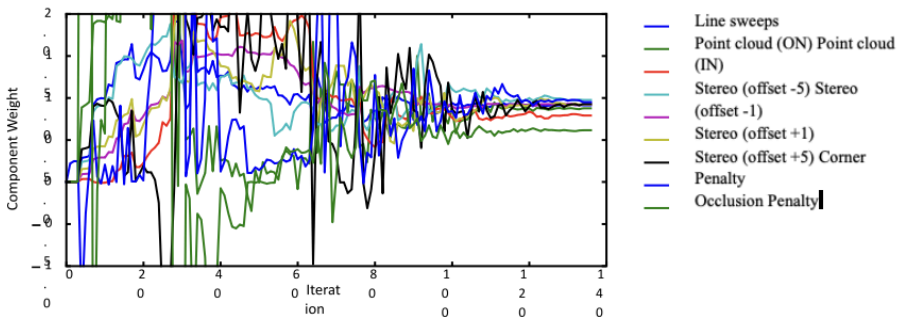

Fig.3: Evolution of w during training. Each series shows the value of one component of w. After an exploration phase the model converges.

found that our learning algorithm assigns small weights to all but the line–sweep features, which are the same features used by [5]. This suggests that the hand– tuned weights are in fact close to optimal within this feature space, though one would expect that with additional feature engineering our learning algorithm would be able to leverage further salient information and reduce the error rate.

7 Discussion

The hypothesis class considered in this paper is among the most complex (in terms of internal constraints on the output space) studied within the structured prediction framework. In this section we turn to some practical lessons learnt that may be of value to other practitioners.

Condition in the joint feature space, not the input feature space. gi A common pre–processing operation for statistical learning is to transform the observed features Θ to zero mean and unit variance. However, for structured prediction tasks it is the joint feature space Ψ that should be conditioned:

$$\Psi' = (\Psi - \mu) / \sigma^2 . \qquad (27)$$

Ideally one would sample from the joint feature space to determine the conditioning transformation, but the distribution of inputs and outputs is generally unknown in an empirical risk minimization setting. Instead, we use the training set as a proxy. We compute the empirical mean and variance of $\{\Psi(\Theta_k, S_k)\}$ at the outset, then apply the transformation (27) after each feature computation.

Condition the loss terms. For any $\eta > 0$, the minimization problem (20) is equivalent (under the substitution $w' = \eta w$, $\xi' = \eta\xi$) to:

$$\min_{(w',\xi')} = \tfrac{1}{2} \|w\|^2 + \eta C \sum_{k=1}^{n} \xi'_k$$

$$s.t.\ \forall k,\ S \neq Sk: \quad \langle w , \Psi(\Theta_k, S_k)\rangle - \langle w' , \Psi(\Theta_k , S)\rangle \geq \eta\Delta(S, S'k) - \xi_k' . \qquad (28)$$

Although any $\eta > 0$ preserves the correctness of the optimization algorithm, we found that choosing $\eta = \mathrm{Var}(\Delta)$ improved numerical stability, since this means the loss terms will have roughly unit variance. Unfortunately, we cannot use the training set to estimate $\mathrm{Var}(\Delta)$ since the loss for the ground truth reconstruction is always zero. Instead we computed $\Delta(\Theta_k, S_j)$ for each $k \neq j$ in the training set. This is not an ideal estimate, but we found that it worked well in practice.

This column represents the predictor trained with respect to Δ_{depth}.

This column represents the predictor trained with respect to $\Delta_{labelling}$.

Check that the hypothesis class contains the ground truth. The algorithm described in [6] implicitly assumes that the hypothesis class Y contains the ground truth labels S_k. This means that if S^+ is the maximizer of (21) then

$$\langle w , \Psi(\Theta_k, S_k)\rangle - \langle w , \Psi(\Theta_k, S^+)\rangle - \Delta(S^+ , S_k) \leq 0 , \qquad (29)$$

since otherwise we would have

$$\langle w , \Psi(\Theta_k, S_k) + \Delta(S_k , S_k)\rangle > \langle w , \Psi(\Theta_k, S^+) + \Delta(S^+ , S_k) , \qquad (30)$$

contradicting S^+ as the maximizer of (21). However, our output space contains fundamentally real–valued quantities such as polygon vertices, which are recovered only to some finite precision by the inference algorithm, and since our ground truth labels were acquired by manual labelling, they sometimes exceed the maximum precision of the inference algorithm. In this case we effectively have $S_k \notin Y$ (although there is always some $S' \in Y$ close to S_k), so it is possible that S^+ violates (29). Our workaround here is simply to check the condition (29) each time we solve the separation problem and, if violated, substitute S_k for S^+. This is justified by the observation that if (29) is violated for S^+ then it is violated for all $S \in Y$. One could think of this as learning with respect to the hypothesis class $Y \cup \{S_k\}$ but evaluating with respect to Y. This is not an ideal solution but we found it to work well in practice. Unfortunately this patch has the side–effect of hiding bugs in the inference algorithm, so care is warranted.

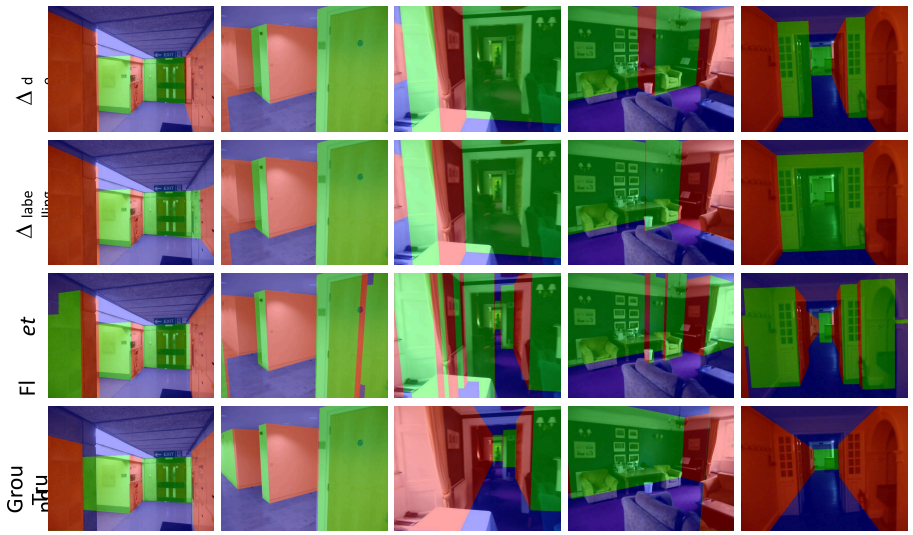

Fig.4: Multiple–view reconstructions predicted by our system (held–out samples). The first two rows represent the predictors trained on Δ_{depth} and $\Delta_{\text{labelling}}$ respectively, the third row is from [4], and the fourth row is ground truth.

8 Conclusion

We have presented a unified learning framework for reconstructing polygonal models from single and multiple views of a scene. We have chosen to work with the indoor Manhattan class of models in order to leverage the parameterization and inference algorithm recently proposed for this hypothesis class [5,4]. Our approach to learning performs a single optimisation with respect to a clearly defined loss function. Experiments show our system out–performing the state–of– the–art for multiple–view reconstruction (by a large margin) and on one metric for single–view reconstruction.

In the future work will extend this approach to learn geometry together with scene classifiers and context–aware object detectors, optimising with respect to a single joint loss function.

References

1. Hoiem, D., Efros, A.A., H´ebert, M.: Geometric context from a single image. In: ICCV. (2005) 654–661
2. Saxena, A., Sun, M., Ng, A.Y.: Make3d: Learning 3d scene structure from a single still image. PAMI **31** (2009) 824–840
3. Lee, D.C., Hebert, M., Kanade, T.: Geometric reasoning for single image structure recovery. In: CVPR. (2009)
4. Flint, A., Reid, I., Murray, D.: Manhattan scene understanding using monocular, stereo, and 3d features. (2011)

5. Flint, A., Mei, C., Reid, I., Murray, D.: A dynamic programming approach to reconstructing building interiors. In: Proc 12th European Conf on Computer Vision. (2010)
6. Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector machine learning for interdependent and structured output spaces. ICML **36** (2004) 104
7. Blaschko, M., Lampert, C.: Learning to localise objects with structured output regression. ECCV (2008) 2–15
8. Taskar, B., Klein, D., Collins, M., Koller, D., Manning, C.: Max-margin parsing. In: Proc. EMNLP. (2004) 1–8
9. Yang, W., Triggs, W., Dai, D., Xia, G.S.: Scene Segmentation with Low Dimensional Semantic Representations and Conditional Random Fields. EURASIP Journal on Advances in Signal Processing **2010** (2010) 1–14
10. Li, Y., Huttenlocher, D.: Learning for stereo vision using the structured support vector machine. In: Proc. ECCV, IEEE (2008) 1–8
11. Hedau, V., Hoiem, D., Forsyth, D.: Recovering the spatial layout of cluttered rooms. In: ICCV. Volume 2. (2009)
12. Marr, D., Poggio, T.: Cooperative computation of stereo disparity. Science **194** (1976) 283–287
13. Scharstein, D., Szeliski, R., Zabih, R.: A taxonomy and evaluation of dense two frame stereo correspondence algorithms. Proceedings IEEE Workshop on Stereo and MultiBaseline Vision SMBV 2001 **47** (2002) 131–140
14. Baillard, C., Baillard, C., Schmid, C., Zisserman, A., Fitzgibbon, A., England, O.O.: Automatic Line Matching And 3D Reconstruction Of Buildings From Multiple Views. ISPRS **32** (1999) 69 – 80
15. Liebowitz, D., Criminisi, A., Zisserman, A.: Creating Architectural Models from Images. Computer Graphics Forum **18** (1999) 39–50
16. Coughlan, J., Yuille, A.: Manhattan world: compass direction from a single image by bayesian inference. In: CVPR. Volume 2. (1999) 941–947 vol.2
17. Furukawa, Y., Curless, B., Seitz, S., Szeliski, R.: Manhattan-world stereo. CVPR **0** (2009) 1422–1429
18. Huffman, D.A.: Impossible objects as nonsense sentences. Machine Intelligence **6** (1971) 295–323
19. Barinova, O., Konushin, V., Yakubenko, A., Lee, K., Lim, H., Konushin, A.: Fast automatic single-view 3-d reconstruction of urban scenes. In: ECCV. (2008) 100–113
20. BakIr, G.H., Hofmann, T., Scholkopf, B., Smola, A.J., Taskar, B., Vishwanathan, S.V.N.: Predicting Structured Data. MIT Press (2007)
21. Joachims, T., Finley, T., Yu, C.N.J.: Cutting-plane training of structural SVMs. Machine Learning **77** (2009) 27–59
22. Hoiem, D., Efros, A.A., Hebert, M.: Automatic photo pop-up. ACM Transactions on Graphics **24** (2005) 577

23. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press, ISBN: 0521540518 (2004)