Indoor Objects and Outdoor Urban Scenes Recognition by 3D Visual Primitives

Junsheng Fu[13] Joni-Kristian Kämäräinen[1] Anders Glent Buch[2]
Norbert Krüger[2]

[1] Vision Group, Tampere University of Technology, Finland, http://vision.cs.tut.fi
[2] CARO Group, University of Southern Denmark, Denmark, http://caro.sdu.dk
[3] Nokia Research Center, Finland

Abstract. Object detection, recognition and pose estimation in 3D images have gained momentum due to availability of 3D sensors (RGB-D) and increase of large scale 3D data, such as city maps. The most popular approach is to extract and match 3D shape descriptors that encode local scene structure, but omits visual appearance. Visual appearance can be problematic due to imaging distortions, but the assumption that local shape structures are sufficient to recognise objects and scenes is largely invalid in practise since objects may have similar shape, but different texture (e.g., grocery packages). In this work, we propose an alternative appearance-driven approach which first extracts 2D primitives justified by Marr's primal sketch, which are "accumulated" over multiple views and the most stable ones are "promoted" to 3D visual primitives. The 3D promoted primitives represent both structure and appearance. For recognition, we propose a fast and effective correspondence matching using random sampling. For quantitative evaluation we construct a semi-synthetic benchmark dataset using a public 3D model dataset of 119 kitchen objects and another benchmark of challenging street-view images from 4 different cities. In the experiments, our method utilises only a stereo view for training. As the result, with the kitchen objects dataset our method achieved almost perfect recognition rate for $\pm 10^\circ$ camera view point change and nearly 90% for $\pm 20^\circ$, and for the street-view benchmarks it achieved 75% accuracy for 160 street-view images pairs, 80% for 96 street-view images pairs, and 92% for 48 street-view image pairs.

1 Introduction

Over the past few decades, object and scene recognition have achieved great success using 2D image processing methods. Recently, with the increasing popularity of Kinect sensors and the emergence of dual-camera mobile phone, researchers are motivated to approach the traditional image recognition problem with 3D computer vision methods. Compared with the successful 2D methods, 3D approaches are not limited to image 2D appearance as the cue for detection and recognition [1, 2]. A number of 3D methods for object and scene recognition have been proposed [3–5] to extract global or local shape descriptors that

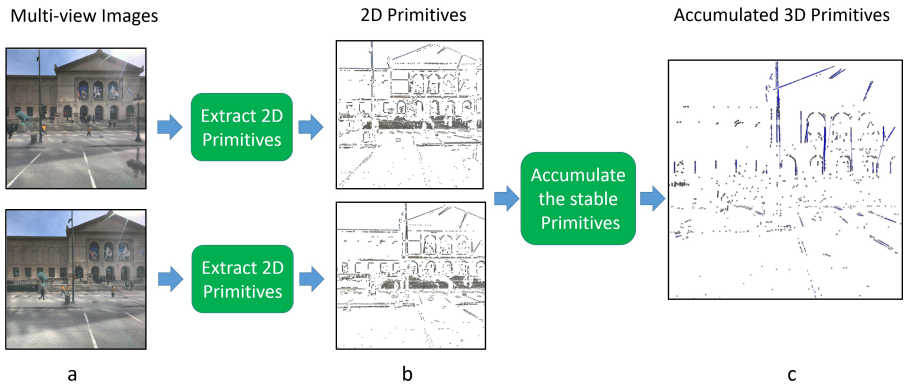

Fig. 1: Construct the 3D primitives from Multi-view images.

encode scene structure, however, they do not take the advantage of 2D visual appearance, e.g. colour and texture.

In accord with the recent trend of 3D object detection and recognition research, we propose in this paper an approach that utilizes both the 2D appearance and 3D structure from the multi-view images. The most important and novel processing of the proposed method, in our view, is the construction of the 3D primitive, i.e. 3D classified features derived from multi-view images. Fig. 1 shows the work-flow of the 3D primitive construction: Firstly, for each multi-view input, the pipeline computes the 2D visual primitives [6] using the intrinsic dimension by Kalkan et al. [7]. Secondly, the stable 2D primitives are matched across multi-view images and triangulated to 3D primitives, as shown in Fig. 1 c (see Section 3 for details). Then the 3D primitives are used for matching 3D objects primitives stored in a database.

To evaluate the proposed method, we tested our pipeline with both indoor objects and outdoor urban scenes. With the indoor objects dataset, our method achieved almost perfect recognition rate for $\pm 10^\circ$ camera view point change and nearly 90% for $\pm 20^\circ$, and for the real world street-view dataset from 4 different cities, our method achieved 75% accuracy for 160 street-view images pairs, 80% for 96 street-view images pairs, and 92 % for 48 street-view image pairs.

Our main contributions are as follows:

- A novel 3D primitive extraction method for object recognition: 2D appearance primitives are extracted and promoted to 3D based on matching results across multi-view images.
- A simple random sampling based recognition to match observed 3D primitives to database objects. The training is based on a single recorded view.
- Novel results on the effect of primitive accumulation vs. no accumulation and 3D matching vs. 2D matching for object recognition in 3D.

– A semi-synthetic benchmark dataset and toolkit of 3D graspable kitchen items captured in the KIT.[1]. This can be used for further analysis in a controlled environment, and the code for rendering novel KIT object views will be made publicly available.
– A real benchmark dataset of stereo street views, which can be used for performances analysis in real conditions.

This paper is structured as follow. Firstly, the related work is presented in Section 2. Then, Section 3 and Section 4 explain the process of constructing 3D primitives from 2D primitives and the matching process of the 3D primitives. Section 5 illustrates the experiment results from both indoor objects database and outdoor street-view images from 4 different cities. Finally, we conclude in Section 6.

2 Related work

The object detection and recognition approaches can be roughly divided into 2D-to-2D (genuine 2D), 3D-to-2D (or 2D-to-3D) and 3D-to-3D (genuine 3D) methods, where the first term defines whether a model (and training data) are 2D or 3D and the latter whether objects are detected from 2D or 3D images. The most successful approach is part-based: local features are extracted and the object described as the parts and their location. Successful results have been reported for detection of visual classes and specific objects in 2D-to-2D [1, 2] and 3D-to-2D [8–10], and many of the methods provide state-of-the-art classification accuracy on common benchmarks.

Our main interest, however, are genuine 3D methods which have not yet reached a mature stage as the aforementioned methods. Next, we give a brief survey on the most recent works, but omit methods based on global description (e.g., [11]), those using temporal information [12, 13] and those tailored for a specific application, such as 3D face recognition [14, 15].

Two notable works related to our method are the ones by Papazov and Burschka [16] and Drost et al. [17]. Papazov and Burschka utilise a random sample principle while Drost et al. use Hough-like voting, but the main commonality is in the fact that they both directly use 3D point clouds, which ties their methods to the selected 3D capturing method. We use local primitives extracted from 2D RGB images. Similar vision primitives were used in Detry et al. [18] ([19]), but their method do not retain 3D structure, and recognition is performed by Markov process message passing utilising pairs of the primitives similar to [17].

The popular 2D interest point detectors and descriptors have also been extended to 3D, for example 3D SURF by Knopp et al. [20], local surface histograms [21] in Pham et al. [22], HOG and DoG by Zaharescu et al. [23] and kernel descriptors [24]. Special 3D shape detectors and descriptors have also been proposed [25, 26] along with neighbourhood processing to improve the robustness of shape descriptors [3, 5]. There are many local 3D shape descriptors (see [27,

[1] http://i61p109.ira.uka.de/ObjectModelsWebUI/

28]), but their main limitation is that they select the points based on local shape information and discard appearance which, after all, is the low-level source of information in the human visual system and used in the Marr's primal sketch [6]. The shape descriptors have been recently evaluated in [4]. One exception is Lee et al. [29] who utilise lines, but that is particularly suitable for their objects of interest (boxes). Hybrids of 3D shape and 2D texture descriptors were proposed by Hu and Zhu [30] and Kang et al. [31].

3 Constructing 3D primitives from 2D primitives

The visual primitives used in this work derive from the primitives found in various layers of the "deep vision hierarchy" [32]. Starting from the pixels (retinal image) we extract low level primitives which are re-sampled (added), deleted, combined (grouped) and promoted through bottom-up processing in the hierarchy. We refer to the operations with a single term, "accumulation". Various computational models of the hierarchy have been proposed [33–35]. out of which we adopt the "cognitive vision model" hierarchy by Pugeault et al. [35]. The main goal of their hierarchy is a symbolic 3D description of a scene, but we form primitives that construct a part-based 3D object model.

On the lowest hierarchy level, 2D primitives are extracted from the left and right images of a stereo pair (see Fig. 1). The primitives are extracted on a regular spatial grid where circular patches are extracted and assigned to one of four low-level classes: a constant colour region, edge/line, junction or texture. The classification is based on computational intrinsic dimensionality [7]. The computational intrinsic dimension, ifD, defined by a real number f measures the effective texture patch dimension similar to the fractal dimension [36], but can be computed fast with linear quadrature filters [37]. The ifD space forms a triangular region where basic perceptual classes map to distinct locations (Fig. 2):

- Constant colour: $ifD \approx i0D$
- Edge/line: $ifD \approx i1D$
- Junction: $i1D << ifD < i2D$
- Textured region $ifD \approx i2D$

The extracted 2D primitives are encoded as

$$\boldsymbol{\pi} = (\boldsymbol{x}, \theta, \phi, \boldsymbol{c}) \tag{1}$$

where $\boldsymbol{x}$ is the 2D image position, θ is the local orientation angle of an edge or line, ϕ is the local phase of an edge/line, and $\boldsymbol{c}$ is the RGB colour vector of the left, middle and right edge colours.

The accumulation of 2D primitives to 3D primitives $\boldsymbol{\Pi}$ is based on multiple views with known calibration: $accumulation : (\boldsymbol{\pi}, \boldsymbol{\pi}') \rightarrow \boldsymbol{\Pi}$. In order to be promoted, the 2D primitive descriptors—colour, orientation and phase—must match, the primitives must lie on their corresponding epipolar lines, and finally the spatial constraints must hold. For putative matches for a primitive $\boldsymbol{\pi}$ at $\boldsymbol{x}$ in the left image, the epipolar line $\boldsymbol{x}' \in l' = \boldsymbol{e}' \times \boldsymbol{H}_\pi \boldsymbol{x}$, where $\boldsymbol{e}' \times \boldsymbol{H}_\pi = \boldsymbol{F}$ is the fundamental matrix [38], in the right image is searched for $\boldsymbol{\pi}'$. Since the

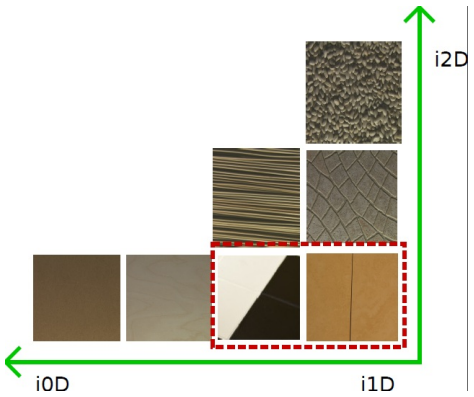

Fig. 2: Texture characterisation in the intrinsic dimension space [7], the 2D line and edge primitives used in our work marked with the dashed line.

2D primitives are computed sparsely on a grid, the matches within the distance of 1.5 times the patch size are accepted. The accumulated 3D primitives are encoded as

$$\boldsymbol{\Pi} = (\boldsymbol{X}, \boldsymbol{n}, \Theta, \Phi, \boldsymbol{C}) \tag{2}$$

where $\boldsymbol{X}$ is the 3D location in space, $\boldsymbol{n}$ is the surface normal, Θ the line/edge orientation, Φ the line/edge phase and $\boldsymbol{C}$ the colour vector constructed by the weighted average of the corresponding 2D colours.

In this work, we use the line/edge primitives (see Fig. 2). The 2D primitive extraction can be adjusted by three quadrature filter parameters [37]. The first parameter is the highest filter frequency (or image resolution). The second parameter is the minimum required energy within the circular patches (normalised to $[0, 1]$) and the third parameter is the maximum variance (normalised to $[0, 1]$), i.e. whether primitives must come from clearly isolated points (low variance). The descriptor match is a weighted sum of colour (weight 0.5), orientation (0.3) and phase (0.06) differences, all normalised to $[0, 1]$, and the match threshold set to 0.3. Moreover, a spatial constraint, "external confidence", similar to stereo algorithms was added to ensure that the accepted 3D primitives are supported by their neighbourhood. By changing the values of the parameters we can affect the number of extracted 2D and 3D primitives and their robustness. Several settings are demonstrated in Table 1 for the first 12 KIT objects.

For the setting 1 approximately 50% of the 2D primitives are promoted. For other settings, the number of 2D primitives is much larger, but due to the accumulation there is not much difference between the number of 3D primitives for the settings 1-3. This is further illustrated in Fig. 3 where the 3D primitives (bottom) look alike for all settings. Note, however, that for Setting 2 and Setting 3 the new primitives are less reliable and therefore more noise appears. By using higher frequencies (a larger image), the number of primitives increases "naturally", i.e., more details are added to places where also the depth informa-

Table 1: Various 3D primitive extraction Parameter settings and the corresponding numbers of produced 3D primitives.

Parameter	*Setting 1*	*Setting 2*	*Setting 3*	*Setting 4*
Image size	300x300	300x300	300x300	400x400
Min. energy	0.4	0.4	0.4	0.4
Max. variance	0.2	0.6	0.2	0.2
Ext. conf.	0.1	0.1	-1.0	0.1

Object	*Setting 1 (2D)*	*Setting 1*	*Setting 2*	*Setting 3*	*Setting 4*
OrangeMarmelade	324	120	243	219	244
BlueSaltCube	410	251	326	315	433
YellowSaltCube	380	201	289	293	338
FruitTea	282	168	258	227	265
GreenSaltCylinder	246	72	158	166	140
MashedPotatoes	424	223	374	329	387
YellowSaltCylinder	355	168	236	247	329
Rusk	503	234	393	303	381
Knaeckebrot	372	186	269	242	300
Amicelli	414	276	384	384	509
HotPot	376	131	200	216	193
YellowSaltCube2	380	210	278	303	396
Avg.	372	187	284	270	326

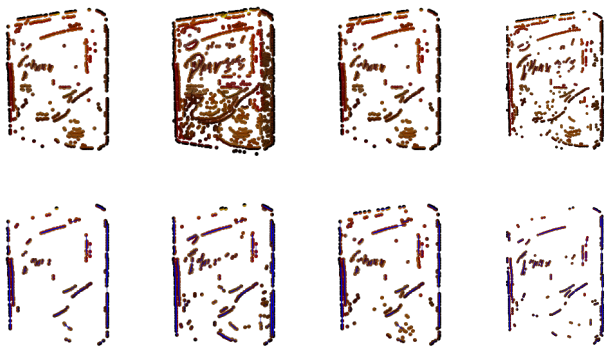

Fig. 3: Top: extracted 2D primitives (stereo left) with Settings 1-4 from the left to right. Bottom: the corresponding 3D primitives after the accumulation.

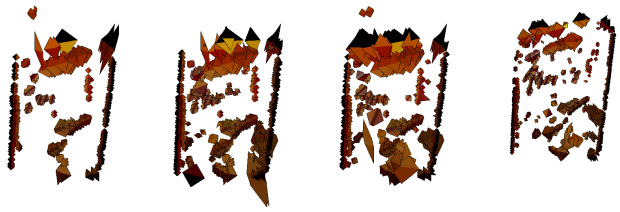

Fig. 4: The 3D primitives at the bottom of Fig. 3 re-drawn using the detected scales. See the last paragraph of Section 3 for details.

tion is reliable. That is illustrated Fig. 4, where 3D primitives are plotted in 3D space with their detected scale.

4 Matching 3D primitives

The 3D primitive based object description in Section 3 represents object appearance in the primitive descriptors Θ, Φ and $\boldsymbol{C}$ and object location in the 3-vectors $\boldsymbol{X}$. The two popular approaches to match descriptors in space are voting and random sampling. A variant of the random sampling appears in Papazov and Burschka [16] and voting (Hough transform) in Drost et al. [17].

The random sampling and voting have certain distinct properties as compared to each other. In the voting approach every primitive is processed once and they cast votes for multiple objects and for multiple poses. The best hypothesis is the one with the highest number of votes. A disadvantage is the size of the vote (accumulator) space, which can become huge without coarse discretisation. In the sampling approach, no accumulation is needed since every random sample generates one hypothesis of an object and its pose. The obvious disadvantage is that the required number of random samples may be large. In other words, the voting is more storage intensive and the sampling more computationally intensive. There exists studies to improve storage requirements and to reduce the number of samples (e.g., [39]), but in this work we select the sampling approach due to its simplicity.

4.1 Random sampling based matching

We randomly sample from the primitives of an object model i (object database), select corresponding primitives from an observed scene, and then compute the transformation $\boldsymbol{T}$ which brings the observed scene and database model primitives in correspondence. The method is similar to Papazov and Burschka [16], except that they directly use dense point cloud points which are sensitive to a selected 3D acquisition process. Additionally, to avoid computational explosion

Algorithm 1 Random sample consensus matching.

1: Compute the match matrix between each observed primitive $\boldsymbol{\Pi}_{i=1\ldots N}$ and each model primitive $\boldsymbol{\Pi}_{i=1\ldots M}$: $\boldsymbol{D}_{N\times M}$.
2: Sort and select the K best matches for each observation primitive $\rightarrow \hat{\boldsymbol{D}}_{N\times K}$.
3: **for** R iterations **do**
4: Randomly select 3 observation primitives from $1\ldots N$ and their correspondences in $1\ldots K$ in $\hat{\boldsymbol{D}}_{N\times K}$.
5: Estimate the linear 3D transformation (isometry/similarity) $\boldsymbol{T}$ using the Umeyama method [40].
6: Transform the all N observation primitives to the model space with $\boldsymbol{T}$.
7: Select the geometrically closest matches (within the K best) and compute the match score s.
8: Update the best match (s_{best}, $\boldsymbol{T}_{best}$) if necessary.
9: **end for**
10: Return s_{best} and $\boldsymbol{T}_{best}$.

(every observation point is a candidate match to every model point), they utilise heuristics. Our method selects the best match using the 3D primitive descriptors. To estimate the 3D transformation (isometry) we use the linear method by Umeyama [40]. A high level algorithm for our matching method is given in Algorithm 1.

There are two important considerations for Algorithm 1: the number of iterations R and a method to compute the match score s. Since the colour plays the most important role in the accumulation, we omit Θ and Φ and use the colour vector $\boldsymbol{C}$ to compute the match matrix $\boldsymbol{D}$. $\boldsymbol{C}$ is a 9-vector of the RGB values for the edge/line left, middle and right which are uniquely defined. The match is the Euclidean distance between the vectors which is fast to compute. Also the colour covariances are available, but using them is computationally inefficient. L^2-normalisation makes the colour descriptors semi illumination invariant.

The number of iterations R is an important parameter since a sufficient number of samples is needed to guarantee that the correct combination is found with high confidence. To derive a formula for R we can consider the ideal case that each N observation point has a correct match in the model. The total number of points is not important, but the number of possible candidates. In Algorithm 1 this is K and we further assume that a correct correspondence is within the K best matches. Now, the probability of randomly selecting a correct combination of three point correspondences (the minimum for 3D isometry/similarity estimation) is

$$P(K) = \frac{1}{K} \cdot \frac{1}{K} \cdot \frac{1}{K} \ . \tag{3}$$

Note that this would be $1/K(K-1)(K-2)$ if the points are shared. The probability that after R iterations no correct triplets have been drawn is $(1-P(K))^R$, and thus, the probability that at least one correct has been drawn is $1-(1-P(K))^R$. The analytical formula for the number of samples in order to pick at least one

correct match with the probability P_S is

$$R = \frac{\log(1 - P_S)}{\log(1 - P(K))} \ . \tag{4}$$

For example, with $P_S = 0.9$ (90% confidence level), we get $R = 287$ for $K = 5$ and $R = 2302$ for $K = 10$. In practise, some primitives have no matches at all, but on the other hand, representation is typically dense in the most informative areas and any primitive near the correct one may succeed. In any case, K should not be more than 10 to limit computational burden ($R \leq 2000$).

To select the best strategy to compute the match score s, we run preliminary tests with the first 12 objects in the KIT dataset (see Table 2 for the results). More details are in Section 5, but here we focus only on the recognition accuracy. The rank order statistics rules, such as *median matching*, are superior due to their robustness to outliers and still computationally affordable. There is no major differences between the median (best 50%) and best 25%, with the number of samples doubled (2× iterations) and isometry vs. similarity, and therefore we selected the median rule. Note that the reverse matching (from models to the scene), is clearly inferior.

Table 2: Recognition accuracies for the first 12 KIT objects using variants of the match score s in Algorithm 1. $K = 10$ best matches and $R = 1000$ random samples (Setting 1, pure chance 8%).

s Method	*El-Az* 5°	*El-Az* 10°	*El-Az* 20°	*El-Az* 30°	*El-Az* 40°
Mean match	84%	74%	50%	34%	18%
Med match	100%	100%	98%	77%	46%
Med match (reverse)	100%	97%	65%	49%	28%
Best25% match	100%	100%	93%	70%	44%
Med match (2× iters)	100%	100%	98%	78%	46%
Med match (simil.)	100%	100%	96%	77%	45%

5 Experiments

In this Section, we evaluate our pipeline with both the **indoor objects dataset** and the **outdoor urban street-view images**.

A dataset was collected in Karlsruhe Institute of Technology (KIT): KIT Object Models Web Database[2]. The KIT dataset provides full high-quality 3D models, so we use the KIT dataset as the indoor objects database for testing the pipeline. For evaluation, we implemented a synthetic view generator that

[2] http://i61p109.ira.uka.de/ObjectModelsWebUI/

can be used to evaluate methods in controlled view points and illumination. To further evaluate the robustness of our pipeline, we gathered 160 street-view images pairs with the known camera poses from 4 different cities. The datasets and experiment results are discussed in the following two Subsections.

5.1 Indoor object dataset

Toolkit for semi-synthetic KIT Objects – The KIT object dataset contains 119 3D captured kitchen items (marmalade packages, mugs, tea packages etc.) suitable for robot grasping and manipulation [41] and stored as high-quality textured 3D polygon models. Using the KIT models (Fig. 5) we provide a public toolkit to generate arbitrary views points, ground truth, and benchmark recognition algorithms.

Fig. 5: Examples of the 119 KIT object models in frontal (training) pose. Note that some objects differ only by details in their appearance (colour or texture).

The toolkit was used to render the training images in roughly frontal pose (Fig. 5), automatically adjust the camera distance to fit objects' bounding boxes to the visible image area, generate stereo pairs (Fig. 6) and output the stereo camera matrices and bounding box world coordinates.

Fig. 6: The stereo pair frontal views of "Amicelli" (left) and "MashedPotatoes" (right). The camera baseline is fixed to 50 world units (1wu ≈ 1mm).

For our experiments, the object database (training set) was made by storing primitives from only one view per object: the frontal views shown in Fig. 5. The test set images were generated by geometrically transforming the same objects by adjusting the camera azimuth and elevation angles. A total of five different test sets were generated using gradually increasing angles: $\{-40°, -30°, \ldots + 40°\}$ This results to 9 test images per object and $119 \times 9 = 1071$ images in total for each test set. The test sets are referred to as Ez-Al-5° ... Ez-Al-40°. The two extremal test set images for an object are illustrated in Fig. 7 and the stereo pairs of each were used to extract the primitives and match them to the all database (training set) objects with Algorithm 1.

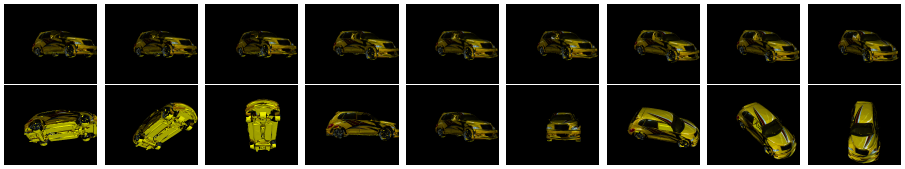

Fig. 7: Variation in the "ToyCarYellow" test images (stereo left): El-Az 5° (top row, the simplest set), and El-Az 40° (bottom, the most difficult set).

Results – The recognition accuracies for all experimental scenarios are presented in Table 3 for the primitive extraction settings Setting 1 and Setting 2 (see Section 3). To compare 2D and 3D matching we utilised directly the 2D primitives with and without the accumulation.

Table 3: Recognition accuracies for the KIT object models (tot. of 1071 test image per set) using median matching (pure chance 0.08%).

Method	*El-Az* 5°	*El-Az* 10°	*El-Az* 20°	*El-Az* 30°	*El-Az* 40°
Med match - Sett. 1	98%	93%	78%	55%	33%
Med match - Sett. 1 (2D)	98%	94%	78%	51%	28%
Med match - Sett. 1 (2D, no acc.)	79%	72%	52%	34%	23%
Med match - Sett. 4	99%	97%	87%	63%	38%
Med match - Shape descr. [42]	88%	75%	47%	33%	19%

Using more primitives achieved by, for example, higher resolution images, is beneficial as the Setting 4 provides the best results. However, the Setting 1 is not significantly worse being much faster (ten seconds vs. minutes in our Matlab implementation). Moreover, the importance of the accumulation process is verified as the 2D matching with accumulated 2D primitives is almost the

same to the accumulated 3D matching. 3D primitives are more beneficial with large view point changes where 2D transformation cannot represent the view anymore.

Overall, for small view angle variation (azimuth and elevation $\leq 10°$) our recognition rate is almost perfect and for 20° still almost 90%. The accuracy starts to drop after 20° due to the fact that the test views start containing structures not present in the training view.

To compare our method with other descriptors, we implemented the local shape context, originally proposed for 2D in [43], extended to 3D by Frome et al. [42] and similar to the heuristic approach in [16]. The local shape context corresponds to a histogram of 3D primitives appearing in the vicinity of each primitive. The local shape context is simple and efficient to compute. The bin size was optimised by cross-validation and the results are shown in the last row of Table 3. For KIT objects, the local shape context descriptors are clearly inferior to the colour matching, but still perform well with the smaller angles and are thus promising for applications and imaging conditions where the colour is not informative.

It is noteworthy that since our approach is genuine 3D it also produces the object pose $\boldsymbol{T}$ as a side product. The detected poses are coarse (Fig. 8), but provide good initial guesses for more accurate pose optimisation.

Fig. 8: Extracted 3D primitives (yellow dots) and database object bounding box and 3D primitives (green) projected by the estimated $\boldsymbol{T}$.

5.2 Outdoor street view scenes

In this part of experiment, 160 street-view image pairs at various locations from 4 different cities were used as benchmark database. These database consists of 40 different urban scenes, where each urban scene has 4 street-view pairs, see Fig. 9 **(a)** as an example.

The ground truth camera pose recorded in the metadata of the street-view images were used to estimate approximate camera extrinsics. For each urban scene, we selected one pair of images for training and the rest 3 pairs for testing. Otherwise, all method settings were the same as in the previous experiment. Without any parameter tuning, we achieved satisfactory results as shown in Table 4.

– For 12 classes (or urban scenes) with 48 street-view pairs, the pipeline achieved 92% accuracy, and 97% of the results ranked the correct class within the 5 best candidates produced by the algorithm.
– For 24 classes with 96 street-view pairs, the pipeline achieved 80% accuracy, and 94% of the results ranked the correct class within the 5 best candidates produced by the algorithm.
– For 40 classes with 160 street-view pairs, the pipeline achieved 75% accuracy, and 85% of the results ranked the correct class within the 5 best candidates produced by the algorithm.

The result shows that our 3D promoted primitives and the simple matching algorithm also work with realistic data of moderate occlusion and viewpoint changes.

Fig. 9: (a) Here are 4 pairs of street-view images for one urban scene. (b) These are 8 examples of urban scenes from our street-view database.

Table 4: Recognition accuracies for outdoor urban scenes using median matching.

Three Sets	*Set1*	*Set2*	*Set3*
Number of classes	12	24	40
Number of street-view pairs	48	96	160
By pure chance to find the correct class	8%	4%	2%
Accuracy	92%	80%	75%
The correct class within the best 5 candidates	97%	94%	85%

6 Conclusions

This paper proposes an approach that utilizes both the 2D appearance and 3D structure from the multi-view images for 3D object detection and recognition. We introduced novel 3D primitives for indoor objects and urban scenes recognition in 3D. The 3D primitive extraction is based on low level visual 2D primitives selected by computational intrinsic dimension that classifies them according to Marr's primal sketch. The 2D primitives are matched across multi-view images and triangulated to 3D primitives. For matching the primitives, we introduced a simple but effective random sampling procedure that achieved 90% accuracy for the view angle variation up to $\pm 20^\circ$ with indoor objects dataset and satisfactory accuracy for the street-view dataset. Our future work will include investigation of other primitive types, such as local texture and higher level primitives, such as constant colour regions.

Acknowledgement. The authors would like to give thanks to Dr. Lixin Fan for the valuable discussions.

References

1. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: CVPR. (2007)
2. Chum, O., Matas, J.: Unsupervised discovery of co-occurrence in sparse high dimensional data. In: CVPR. (2010)
3. Rodola, E., Albarelli, A., Bergamasco, F., Torsello, A.: A scale independent selection process for 3d object recognition in cluttered scenes. Int J Comput Vis **102** (2013) 129–145
4. As'ari, M., Supriyanto, U.S.E.: 3d shape descriptor for object recognition based on kinect-like depth image. Image and Vision Computing **32** (2014) 260–269
5. Buch, A., Yang, Y., Krüger, N., Petersen, H.: In search of inliers: 3d correspondence by local and global voting. In: CVPR. (2014)
6. Marr, D.: Vision. A Computational Investigation into the Human Representation and Processing of Visual Information. W.H. Freeman and Company (1982)
7. Kalkan, S., Wörgötter, F., Krüger, N.: Statistical analysis of local 3d structure in 2d images. In: CVPR. (2006)

8. Glasner, D., Galun, M., Alpert, S., Basri, R., Shakhnarovich, G.: Viewpoint-aware object detection and pose estimation. In: ICCV. (2011)
9. Sattler, T., Leibe, B., Kobbelt, L.: Fast image-based localization using direct 2d-to-3d matching. In: ICCV. (2011)
10. Zia, M., Stark, M., Schiele, B., Schindler, K.: Detailed 3d representations for object recognition and modeling. IEEE PAMI **35** (2013)
11. Dorai, C., Jain, A.: Shape spectrum based view grouping and matching of 3D free-form objects. T-PAMI **19** (1997)
12. Fayad, J., Russell, C., Agapito, L.: Automated articulated structure and 3D shape recovery from point correspondences. In: ICCV. (2011)
13. Sharma, A., Horaud, R., Cech, J., Boyer, E.: Topologically-robust 3D shape matching based on diffusion geometry and seed growing. In: CVPR. (2011)
14. Bronstein, A., Bronstein, M., Kimmel, R.: Three-dimensional face recognition. Int J Comput Vis **64** (2005)
15. Gökberg, B., Irfanoglu, M., Akarun, L.: 3D shape-based face representation and feature extraction for face recognition. Image and Vision Computing **24** (2006)
16. Papzov, C., Burschka, D.: An efficient RANSAC for 3D object recognition in noisy and occluded scenes. In: ACCV. (2010)
17. Drost, B., Ulrich, M., Navab, N., Ilic, S.: Model globally, match locally: Efficient and robust 3D object recognition. In: CVPR. (2010)
18. Detry, R., Pugeault, N., Piater, J.: A probabilistic framework for 3D visual object representation. T-PAMI **31** (2009)
19. Baseski, E., Pugeault, N., Kalkan, S., Kraft, D., Wörgötter, F., Krüger, N.: A scene representation based on multi-modal 2d and 3d features. In: ICCV Workshop on 3D Representation for Recognition. (2007)
20. Knopp, J., Prasad, M., Willems, G., Timofte, R., van Gool, L.: Hough transform and 3D SURF for robust three dimensional classification. In: ECCV. (2010)
21. Tombari, F., Salti, S., Stefano, L.D.: Unique signatures of histograms for local surface description. In: ECCV. (2010)
22. Pham, M.T., Woodford, O., Perbert, F., Maki, A., Stenger, B., Cipolla, R.: A new distance for scale-invariant 3D shape recognition and registration. In: ICCV. (2011)
23. Zaharescu, A., Boyer, E., Horaud, R.: Keypoints and local descriptors of scalar functions on 2d manifolds. Int J Comput Vis **100** (2012) 78–98
24. Bo, L., Lai, K., Ren, X., Fox, D.: Object recognition with hierarchical kernel descriptors. In: CVPR. (2011)
25. Sun, J., Ovsjanikov, M., Guibas, L.: A concise and provably informative multi-scale signature based on heat diffusion. In: Eurographics Symposium on Geometry Processing. (2009)
26. Bronstein, A., Bronstein, M., Guibas, L., Ovsjanikov, M.: Shape google: Geometric words and expressions for invariant shape retrieval. ACM Trans. on Graphics (2011)
27. Ahmed, N., Theobalt, C., Rössl, C., Thrun, S., Seidel, H.P.: Dense correspondence finding for parameterization-free animation reconstruction from video. In: CVPR. (2008)
28. Mian, A., Bennamoun, M., Owens, R.: On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes. Int J Comput Vis **89** (2010) 348–361
29. Lee, S., Lu, Z., Kim, H.: Probabilistic 3D object recognition with both positive and negative evidences. In: ICCV. (2011)
30. Hu, W., Zhu, S.C.: Learning a probabilistic model mixing 3d and 2d primitives for view invariant object recognition. In: CVPR. (2010)

31. Kang, H., Hebert, M., Kanade, T.: Discovering object instances from scenes of daily living. In: ICCV. (2011)
32. Krüger, N., Janssen, P., Kalkan, S., Lappe, M., Leonardis, A., Piater, J., Rodriguez-Sanchez, A., Wiskott, L.: Deep hierarchies in the primate visual cortex: What can we learn for computer vision? IEEE PAMI **35** (2013)
33. Fidler, S., Boben, M., Leonardis, A.: Similarity-based cross-layered hierarchical representation for object categorization. In: CVPR. (2008)
34. Mutch, J., Lowe, D.: Object class recognition and localization using sparse features with limited receptive fields. Int J Comput Vis **80** (2008) 45–57
35. Pugeault, N., Wörgötter, F., Krüger, N.: Accumulated visual representation for cognitive vision. In: BMVC. (2008)
36. Chaudhuri, B., Sarkar, N.: Texture segmentation using fractal dimension. T-PAMI **17** (1995)
37. Felsberg, M., Sommer, G.: Image features based on a new approach to 2D rotation invariant quadrature filters. In: ECCV. (2002)
38. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. (2003)
39. Chum, O., Matas, J.: Optimal randomized RANSAC. T-PAMI **30** (2008)
40. Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns. T-PAMI **13** (1991)
41. Xue, Z., Kasper, A., Zoellner, J., Dillmann, R.: An automatic grasp planning system for service robots. In: ICAR. (2009)
42. Frome, A., Huber, D., Kolluri, R., Bulow, T., Malik, J.: Recognizing objects in range data using regional point descriptors. In: ECCV. (2004)
43. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape context. T-PAMI **24** (2002)