Computer Vision and Image Understanding 117 (2013) 1190–1202

Contents lists available at SciVerse ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Object detection, shape recovery, and 3D modelling by depth-encoded hough voting ☆

Min Sun [a,*], Shyam Sunder Kumar [a], Gary Bradski [b], Silvio Savarese [a]

[a] Department of Electrical and Computer Engineering, The University of Michigan, Ann Arbor, MI 48105, United States
[b] Founder/Chief Scientist at Industrial Perception Inc., CA, United States

ARTICLE INFO

Article history:
Received 24 July 2012
Accepted 10 May 2013
Available online 22 May 2013

Keywords:
Object recognition
Object detection
Viewpoint estimation
Shape recovery
3D reconstruction
Shape completion
Texture completion

ABSTRACT

Detecting objects, estimating their pose, and recovering their 3D shape are critical problems in many vision and robotics applications. This paper addresses the above needs using a two stages approach. In the first stage, we propose a new method called DEHV – Depth-Encoded Hough Voting. DEHV jointly detects objects, infers their categories, estimates their pose, and infers/decodes objects depth maps from either a single image (when no depth maps are available in testing) or a single image augmented with depth map (when this is available in testing). Inspired by the Hough voting scheme introduced in [1], DEHV incorporates depth information into the process of learning distributions of image features (patches) representing an object category. DEHV takes advantage of the interplay between the scale of each object patch in the image and its distance (depth) from the corresponding physical patch attached to the 3D object. Once the depth map is given, a full reconstruction is achieved in a second (3D modelling) stage, where modified or state-of-the-art 3D shape and texture completion techniques are used to recover the complete 3D model. Extensive quantitative and qualitative experimental analysis on existing datasets [2–4] and a newly proposed 3D table-top object category dataset shows that our DEHV scheme obtains competitive detection and pose estimation results. Finally, the quality of 3D modelling in terms of both shape completion and texture completion is evaluated on a 3D modelling dataset containing both in-door and out-door object categories. We demonstrate that our overall algorithm can obtain convincing 3D shape reconstruction from just one single uncalibrated image.

Published by Elsevier Inc.

1. Introduction

Detecting objects and estimating their geometric properties are crucial problems in many application domains such as robotics, autonomous navigation, high-level visual scene understanding, surveillance, gaming, object modelling, and augmented reality. For instance, if one wants to design a robotic system for grasping and manipulating objects, it is of paramount importance to encode the ability to accurately estimate object orientation (pose) from the camera view point as well as recover structural properties such as its 3D shape. This information will help the robotic arm grasp the object at the right location and successfully interact with it. Moreover, if one wants to augment the observation of an environment with virtual objects, the ability to reconstruct visually pleasing 3D models for object categories is very important.

This paper addresses the above needs, and tackles the following challenges: (i) Learn models of object categories by combining view specific depth maps along with the associated 2D image of object instances of the same class from different vantage points. Depth maps with registered RGB images can be easily collected using sensors such as Kinect Sensor [5]. We demonstrate that combining imagery with 3D information helps build richer models of object categories that can in turn make detection and pose estimation more accurate. (ii) Design a coherent and principled scheme for detecting objects and estimating their pose from either just a single image (when no depth maps are available in testing) (Fig. 1b), or a single image augmented with depth maps (when these are available in testing). In the latter case, 3D information can be conveniently used by the detection scheme to make detection and pose estimation more robust than in the single image case. (iii) Have our detection scheme reconstruct the 3D model of the object from just a single uncalibrated image (when no 3D depth maps are available in testing) (Fig. 1c–g) and without having seen the object instance during training.

In this paper, we propose a two stages approach to address the above challenges (Fig. 2). In the first stage, our approach seeks to (i) detect the object in the image, (ii) estimate its pose, and (iii) recover a rough estimate of the object 3D structure (if no depth maps are available in testing). This is achieved by introducing a new

☆ This paper has been recommended for acceptance by Carlo Colombo.

* Corresponding author.
E-mail addresses: sunmin@umich.edu (M. Sun), shyamsk@umich.edu (S.S. Kumar), garybradski@gmail.com (G. Bradski), silvio@eecs.umich.edu (S. Savarese).

1077-3142/$ - see front matter Published by Elsevier Inc.
http://dx.doi.org/10.1016/j.cviu.2013.05.002

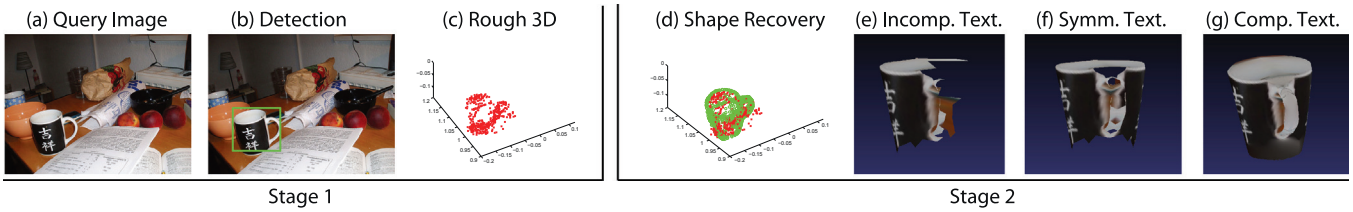

Fig. 1. Key steps of our reconstruction algorithm: (a) Single query 2D image. (b) Detected object; the bounding box indicates the location where the object has been estimated in the image; Our proposed Depth Encoded Hough Voting (DEHV) detector can be used to recognize object class label, roughly estimate the object pose (i.e., object orientation in the camera reference system), and automatically reconstructs surface elements (3D points) in the camera reference system (c). As figure shows, the reconstruction is clearly partial and incomplete; (d) Shape recovery: by using the estimated object class label and pose, we propose a novel 2D + 3D ICP algorithm to register the reconstructed surface elements with one of the 3D models that is available in training; this allows to infer the object 3D structure in regions that are not visible from the query image. (e) Texture mapping: after performing 3D shape registration, we texture map image texture to the 3D shape model; again, the object texture is incomplete as we cannot map image texture to occluded surface elements. (f) Texture completion: we use the fact that some object categories are symmetric to transfer image texture to the occluded regions. (g) Remaining un-textured surfaces elements are completed using image compositing methods inspired by [6].

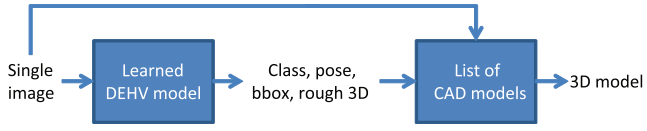

Fig. 2. Flow chart showing the process of our proposed system.

formulation of the Implicit Shape Model (ISM) [1] and generalized Hough voting scheme [7]. In our formulation, depth information is incorporated into the process of learning distributions of object image patches that are compatible with the underlying object location (shape) in the image plane. We call our scheme *DEHV – Depth-Encoded Hough Voting scheme* (Section 3.1). DEHV addresses the intrinsic weaknesses of existing Hough voting schemes [1,8–10] where errors in estimating the scale of each image object patch directly affects the ability of the algorithm to cast consistent votes for the object existence. To resolve this ambiguity, we take advantage of the interplay between the scale of each object patch in the image and its distance (depth) from the corresponding physical patch attached to the 3D object, and specifically use the fact that objects (or object parts) that are closer to the camera result in image patches with larger scales. Depth is encoded in training by using available depth maps of the object from a number of view points. At recognition time, DEHV is applied to detect objects (Fig. 1b), estimate their pose, and simultaneously infer their 3D structure given hypotheses of detected objects (Fig. 1c). The object 3D structure is inferred at recognition time by estimating (decoding) the depth (distance) of each image patch involved in the voting from the camera center. Critically, depth decoding can be achieved even if just a single test image is provided. If depth maps are available in testing, the additional information can be used to further validate if a given detection hypothesis is correct or not. We summarize the inferred quantities in Fig. 3 and the required supervision in Fig. 4. Notice that the inferred object 3D structure

Single Image		
	Depth in testing	No depth in testing
Inferred quantities	object class, location, scale, pose	object class, location, scale, pose, depth map

Fig. 3. Estimated quantities in Stage 1.

Stage 1	Stage 2
-Images of object from multiple views -Depth maps of object from multiple views -Bounding boxes and pose annotation	-List of CAD Models

Fig. 4. Required degree of supervision in training for each stage.

from stage one is partial (it does not account for the portions of the object that are not visible from the query image) and sparse (it only recovers depth for each voting patch). The goal of the second stage is to obtain a full 3D object model where both 3D structure and albedo properties (texture) are also recovered.

In the second stage, the information inferred from stage one (object location in the image, scale, pose, and rough 3D structure) is used to obtain a full 3D model of the object. Specifically, we consider a 3D modelling stage where a full 3D model of the object is obtained by 3D shape recovery and texture completion (Section 3.2). We carry out 3D shape recovery (i.e., infer shape from the unseen regions) by: (i) utilizing 3D shape exemplars from a database of 3D CAD models which can be collected from [11] and other online 3D warehouses, or obtained by shape from silhouette [12] and (ii) applying a novel 2D + 3D iterative closest point (ICP) matching algorithm which jointly registers the best 3D CAD model to the inferred 3D shape and the occlusion boundaries of back projected 3D CAD model to object contours in the image. By choosing the best fit, our system obtains a plausible full reconstruction of the object 3D shape (Section 3.3) (Fig. 1d). Object appearance is rendered by texture mapping the object image into the 3D shape. Such texture is clearly incomplete as non-visible object surface areas cannot be texture mapped (Fig. 1e). Thus, we perform texture completion by: (i) transferring texture to such non-visible object surface areas by taking advantage of the fact that some object categories are symmetric (when possible) (Fig. 1f) and (ii) using an error-tolerant image compositing technique inspired by [6] to fill the un-textured regions (i.e., holes) (Section 3.4) (Fig. 1g). We summarize the required supervision in Fig. 4.

Extensive experimental analysis on a number of public datasets (including car Pascal VOC07 [2], mug ETHZ Shape [3], mouse and stapler 3D object dataset [13]), an two in-house datasets (comprising at most five object categories), where ground truth 3D information is available, are used to validate our claims (Section 4). Experiments with the in-house datasets demonstrate that our

DEHV scheme: (i) achieves better detection rates (compared to the traditional Hough voting scheme); further improvement is observed when depth maps are available in testing; (ii) produces convincing 3D reconstructions from single images; the accuracy of such reconstructions have been qualitatively assessed with respect to ground truth depth maps; (iii) achieves accurate 3D shape recovery and visually pleasing texture completion results. Experiments with public datasets demonstrate that our DEHV successfully scales to different types of categories and works in challenging conditions (severe background clutter, occlusions). DEHV achieves state of the art detection results on several categories in ETHZ Shape dataset [3], and competitive pose estimation results on 3D object dataset [13]. We also evaluate the accuracy of shape completion and quality of the texture completion on the 3D modelling dataset (Section 3.2). Finally, we show typical results demonstrating that DEHV is capable to produce convincing 3D reconstructions from single uncalibrated images using Pascal VOC07 dataset [2], ETHZ Shape dataset [3], and 3D object dataset [13] in Figs. 19 and 15.

2. Previous work

In the last decade, the vision community has made substantial progress addressing the problem of object categorization from 2D images. While most of the work has focussed on representing objects as 2D models [14,1,15] or collections of 2D models [16], very few methods have tried to combine in a principled way the appearance information that is captured by images and the intrinsic 3D structure of an object category. Works by [17,13,4] have proposed solutions for modelling the way how 2D local object features (or parts) and their relationship vary in the image as the camera view point changes. Other works [18–21] propose hybrid representations where reconstructed 3D object models are augmented with features or parts capturing diagnostic appearance. Interestingly, few of these methods [22,21,23] have demonstrated and evaluated the ability to recover 3D shape information from a single query image. The work by [22] is the closest to ours in spirit. Authors in [22] propose to use image patches to transfer metadata (i.e., depth). Unlike our method, 3D information is not directly encoded into the model during training. Other works propose to address the problem of detecting and estimating geometrical properties of single object instances [24–27]; while accurate pose estimation and 3D object reconstruction are demonstrated, these methods cannot be easily extended to incorporate intra-class variability so as to detect and reconstruct object categories. Unlike our work, these techniques also require that the objects have sufficient "texture" on their surface to carry out successful geometric registration. Other approaches assume that additional information about the object is available in both training and testing (videos, 3D range data) [28,29]. These approaches tend to achieve high detection accuracy and pose estimation relying on more expensive hardware platforms, and fail when the additional 3D data is either partially or completely unavailable. A comprehensive survey of 3D object detection method is presented in [30].

2.1. 3D modelling

Research on 3D object and scene modelling from images has received a large amount of attention in the graphics and vision community. Such approaches are often referred to as image-based modelling techniques (IBM). Starting from early work by [31,32], IBM techniques have been recently employed for successfully modelling large scale environments such as city environments from large collection of images on the internet [33,34]. IBM techniques often require different degrees of human intervention [31] or the assumptions that special equipments are available and/or cameras are calibrated [35,36].

Even if outstanding results have been produced, many of these methods make the basic assumption that several images (portraying the object in the scene from different view points) are available. However, this is not always the case. Recovering scene geometry from a single view has been initially explored under the assumption of having users guiding the reconstruction [37,38] or augmenting the photograph with additional 3D data [39]. Recently, researchers have proposed to apply machine learning methodologies for resolving the 3D-2D mapping ambiguity and obtaining convincing reconstructions of outdoor [40,41] and indoor scenes [42–45] from just one single image.

Alternative techniques have been proposed for modelling specific 3D objects (rather than scenes or environments). Again, depending on the application and the level of accuracy that one aims to achieve, researchers have proposed methods employing either external lighting sources such as lamps [46,47], projectors [48], lasers [49], or a number of calibrated [50] or uncalibrated views obtained using external devices such as turntables [51]. A recent survey nicely summarizes most relevant works [52] from an almost endless literature on this topic. Recently, Prasad et al. [53] have proposed a method to reconstruct deformable object classes from multiple and unordered images. Due to the absence of reliable point correspondences across deformable object instances, class-specific curve correspondences need to be manually selected.

The reconstruction of an underlying 3D shape model is not always a necessary step if one wants to render the environment appearance from just images. These methods fall under the name of image based rendering approaches (IBR). Works by [54–57] are among the most notable examples. The lack of the underlying 3D shape model, however, makes it harder for these techniques to be used in applications where virtual worlds are to be augmented with the reconstructed models.

As opposed to indoor or outdoor scenes where cues such as vanishing lines or texture foreshortening are available, fewer methods have been proposed for recovering 3D models of objects from a single image. Researchers mostly focused on recovering 3D shape models from object contours (silhouettes) extracted or identified on a single image either automatically [58,59] or through some level of user intervention [60–62]. These methods, however, often assume topological properties of objects such as smoothness, convexity, or cylindrical symmetry or heavily relies on user intervention. In our work, we do not want our query objects to be subject to these constraints. Rather, similar to [40,41], we advocate the usage of machine learning for solving the daunting task of single view object reconstruction with arbitrary topology and minimal user intervention. Very recently, [63,64,21,23] have shown the ability to reconstruct sparse/partial 3D object points from a single image. However, none of these methods have been extensively tested so as to demonstrate that realistic 3D models of objects can be obtained.

3. Our method

To summarize, our method can be roughly decomposed in a recognition/reconstruction stage and a 3D modelling stage.

In the recognition/reconstruction stage, Depth-Encoded-Hough-Voting detectors (DEHV) [64], trained with both object 3D shape and local diagnostic appearance information, identifies object' locations and classes, and recovers approximate and partial 3D structure information from a single query image (Section 3.1) (Fig. 1(a-c)).

Because we obtain only a partial reconstruction (object surface that is not visible from the query image cannot be reconstructed at

this stage). Thus, we consider a 3D modelling stage where a full 3D model of the object is obtained by 3D shape recovery and texture completion (Section 3.2) (Fig. 1d–g).

3.1. Stage 1: Depth-Encoded Hough Voting

In recognition techniques based on hough voting [7] the main idea is to represented the object as a collection of parts (patches) and have each part to cast votes in a discrete voting-space. Each vote corresponds to a hypothesis of object location x and class O. The object is identified by the conglomeration of votes in the voting space $V(O,x)$. $V(O,x)$ is typically defined as the sum of independent votes $p(O,x,f_j,s_j,l_j)$ from each part j, where l_j is the location of the part, s_j is the scale of the part, and f_j is the part appearance.

Previously proposed methods [1,8–10] differ mainly by the mechanism for selecting good parts. For example, parts may be either selected by an interest point detector [1,9], or densely sampled across many scales and locations [8]; and the quality of the part can be learned by estimating the probability [1] that the part is good or discriminatively trained using different types of classifiers [9,8]. In this paper, we propose a novel method that uses 3D depth information to guide the part selection process. As a result, our constructed voting space $V(O,x|D)$, which accumulates votes for different object classes O at location x, depends on the corresponding depth information D of the image. Intuitively, any part that is selected at a wrong scale can be pruned out by using depth information. This allows us to select parts which are consistent with the object physical scale. It is clear that depending on whether object is closer or further, or depending on the actual 3D object shape, the way how each patch votes will change (Fig. 5).

In detail, we define $V(O,x|D)$ as the sum of individual probabilities over all observed images patches at location l_j and for all possible scales s_j, i.e,

$$\begin{aligned} V(O,x|D) &= \sum_j \int p(O,x,f_j,s_j,l_j|d_j)\ ds_j \\ &= \sum_j \int p(O,x|f_j,s_j,l_j,d_j)p(f_j|s_j,l_j,d_j) \\ &\quad p(s_j|l_j,d_j)P(l_j|d_j)\ ds_j \end{aligned} \tag{1}$$

where the summation over j aggregates the evidence from individual patch location, and the integral over s_j marginalizes out the uncertainty in scale for each image patch. Since f_j is calculated deterministically from observation at location l_j with scale s_j, and we assume $p(l_j|d_j)$ is uniformly distributed given depth, we obtain:

$$\begin{aligned} V(O,x|D) &\propto \sum_j \int p(O,x|f_j,s_j,l_j,d_j)p(s_j|l_j.d_j)ds_j \\ &= \sum_{j,i} \int p(O,x|C_i,s_j,l_j,d_j)p(C_i|f_j) \\ &\quad p(s_j|l_j,d_j)ds_j \end{aligned} \tag{2}$$

Here we introduce codebook entry C_j, matched by feature f_j, into the framework, so that the quality of a patch selected will be related to which codeword it is matched to. Noting that C_j is calculated only using f_j and not the location l_j, scale s_j, and depth d_j, we simplify $p(C_j|f_j,\ s_j,\ l_j,\ d_j)$ into $p(C_j|f_j)$. And by assuming that $p(O,x|.)$ does not depend on f_j given C_j, we simplify $p(O,x|C_j,f_j,s_j,l_j,d_j)$ into $p(O,x|C_j,s_j,l_j,d_j)$.

Finally, we decompose $p(O,x|.)$ into $p(O|.)$ and $p(x|.)$ as follows:

$$\begin{aligned} V(O,x|D) &\propto \sum_{j,i} \int p(x|O,C_i,s_j,l_j,d_j)p(O|C_i,s_j,l_j,d_j) \\ &\quad p(C_i|f_j)p(s_j|l_j,d_j)\ ds_j \end{aligned} \tag{3}$$

3.1.1. Interplay between scale and depth

We design our method so as to specifically selects image patches that tightly enclose a sphere with a fix radius r in 3D during training. As a result, our model enforces a 1-to-1 mapping m between scale s and depth d. This way, given the 3D information, our method deterministically select the scale of the patch at each location l, and given the selected patches, our method can infer the underlying 3D information (Fig. 6). In detail, given the camera focal length t, the corresponding scale s at location $l=(u,v)$ can be computed as $s=m(d,l)$ and the depth d can be inferred from $d=m^{-1}(s,l)$. The mapping m obeys the following relations:

$$\begin{aligned} &s = 2(\bar{v}-v);\ \ \bar{v} = \tan(\theta+\phi)t \\ &\theta = \arcsin\left(\frac{r}{d_{yz}}\right);\ \ \phi = \arctan\left(\frac{v}{t}\right) \\ &d_{yz} = \frac{d\sqrt{t^2+v^2}}{\sqrt{u^2+v^2+t^2}}:\ d \text{ projected onto } yz \text{ plane} \end{aligned} \tag{4}$$

Hence, $p(s|l,d)=\delta(s-m(d,l))$. Moreover, using the fact that there is a 1-to-1 mapping between s and d, probabilities $p(x|.)$ and $p(O|.)$ are independent to d given s. As a result, only scale s is directly influenced by depth.

In the case when depth is unknown, $p(s|l,d)$ becomes a uniform distribution over all possible scales. Our model needs to search through the scale space to find patches with correct scales. This will be used to detect the object and simultaneously infer the depth $d=m^{-1}(s,l)$. Hence, the underlying 3D shape of the object will be recovered.

3.1.2. Random forest codebook

In order to utilize dense depth map or infer dense reconstruction of an object, we use random forest to efficiently map features f into codeword C (similar to [8]) so that we can evaluate patches densely distributed over the object. Moreover, random forest is discriminatively trained to select salient parts. Since feature f deterministically maps to C^i given the i_{th} random tree, the voting score $V(O,x|D)$ becomes:

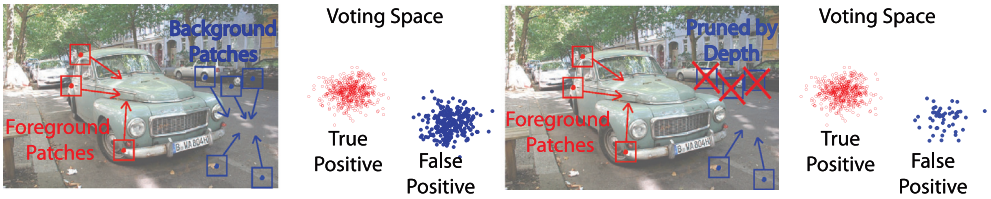

Fig. 5. Right panel shows that patches associated to the actual object parts (red boxes) will vote for the correct object hypothesis (red dots) in the voting space on the right. However, parts from the background or other instances (cyan boxes) will cast votes that may create a false object hypothesis (green dots) in the voting space. Left panel shows that given depth information, the patches selected at a wrong scale can be easily pruned. As a result, the false positive hypothesis will be supported by less votes. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

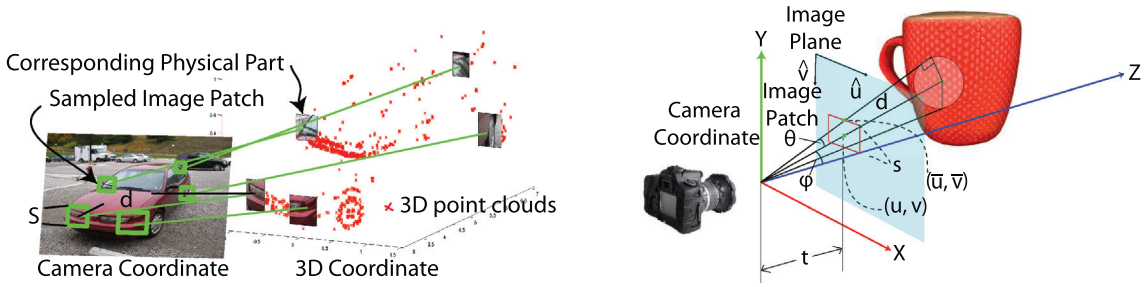

Fig. 6. Illustration of interplay between scale and depth (depth to scale mapping). Top panel illustrates the interplay between scale and depth. We make the assumption that an image patch (green box) tightly encloses the physical 3D part with a fix size. During training, our method deterministically selects patches given the patch center l, 3D information of the image, and focal length t. During testing, given the selected image patches on the object, our method directly infers the location of the corresponding physical parts and obtains the 3D shape of the object. Bottom Panel illustrates the physical interpretation of Eq. (4). Under the assumption that image patch (red bounding box) tightly encloses the 3D sphere with radius r, the patch scale s is directly related to the depth d given camera focal length t and the center $l = (u, v)$ of the image patch. Notice that this is a simplified illustration where the patch center is on the yz plane. This figure is best viewed in colour. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

$$V(O,x|D) \propto \sum_{j,i} \int p(x|O,C^i(f_j),s_j,l_j)p(O|C^i(f_j)) \\ p(s_j|l_j,d_j)ds_j \tag{5}$$

where the summation over i aggregates the discriminative strength of different trees. In Section 3.1.3, we describe how the distributions of $p(x|O,C^i(f_j),s_j,l_j)$ and $p(O|C^i(f_j))$ are learned given training data, so that each patch j knows where to vast votes during recognition.

3.1.3. Training the model

We assume that for a number of training object instances, the 3D reconstruction D of the object is available. This corresponds to having available the distance (depth) of each image object patch from its physical location in 3D. Our goal is to learn the distributions of location $p(x|.)$ and object class $p(O|.)$, and the mapping of $C^i(f)$. Here we define location x of an object as a bounding box with center position q, height h, and aspect ratio a. We sample each image patch centered at location l and select the scale $s = m(l,d)$. Then the feature f is extracted from the patch (l,s). When the image patch comes from a foreground object, we cache: (1) the information of the relative voting direction b as $\frac{q-l}{s}$; (2) the relative object-height/patch-scale ratio w as $\frac{h}{s}$; (3) the object aspect ratio a. Then, we use both the foreground patches (positive examples) and background patches (negative examples) to train a random forest to obtain the mapping $C^i(f)$. $p(O|C)$ is estimated by counting the frequency that patches of O falls in the codebook entry C. $p(x|O,C,s,l)$ can be evaluated given the cached information $\{v,w,a\}$ as follows:

$$p(x|O,C,s,l) \propto \sum_{j \in g(O,C)} \delta(q - b_j \cdot s + l, h - w_j \cdot s, a - a_j)$$

where $g(O,C)$ is a set of patches from O mapped to codebook entry C.

3.1.4. Recognition and 3D reconstruction

3.1.4.1. Recognition when depth is available. It is straightforward to use the model when 3D information is observed during recognition. Since the uncertainty of scale is removed, Eq. (5) becomes

$$V(O,x|D) \propto \sum_{j,i} p(x|O,C^i(f_j),m(l_j,d_j),l_j)p(O|C^i(f_j))$$

Since $s_j = m(l_j,d_j)$ is a single value at each location j, the system can detect objects more efficiently by computing less features and counting less votes. Moreover, patches selected using local appearance at a wrong scale can be pruned out to reduce hallucination of objects (Fig. 5).

3.1.4.2. Recognition when depth is not available. When no 3D information is available during recognition, $p(s_j|l_j,d_j)$ becomes a uniform distribution over the entire scale space. Since there is no closed form solution of integral over s_j, we propose to discretize the space into a finite number of scales S so that Eq. (5) can be approximated by

$$V(O,x|D) \propto \sum_{j,i} \sum_{s_j \in S} p(x|O,C^i(f_j),s_j,l_j)p(O|C^i(f_j)).$$

3.1.4.3. Decoding 3D information. Once we obtain a detection hypothesis (x,O) (green box in Fig. 7a) corresponding to a peak in the voting space V, the patches that have cast votes for a given hypothesis can be identified (red cross in Fig. 7a). Since the depth information is encoded by the scale s and position l of each image patch, we apply Eq. (4) in a reverse fashion to infer/decode depths from scales. The reconstruction, however, is affected by a number of issues: (i) *Quantization error*: The fact that scale space is discretized into a finite set of scales, implies that the depths d that we obtained are also discretized. As a result, we observe the reconstructed point clouds as slices of the true object (see Fig. 7b). We propose to use the height of the object hypothesis h and the specific object-height/patch-scale ratio w to recover the continuous scale $\hat{s} = h/w$. Notice that since w is not discretized, $\hat{s}$ is also not discretized. Hence, we recover the reconstruction of an object as a continuum of 3D points (see Fig. 7c). (ii) *Phantom objects*: The strength and robustness of our voting-based method comes from the ability to aggregate pieces of information from different training instances. As a result, the reconstruction may contain multiple phantom objects since image patches could resemble those coming from different training instances with slightly different intrinsic scales. Notice that the phantom objects phenomenon reflects the uncertainty of the scale of the object in an object categorical model. In order to construct a unique shape of the detected object instance, we calculate the relative object height in 3D with respect to a selected reference instance to normalize the inferred depth. Using this method, we infer a unique 3D structure of the visible surface of the detected object.

3.1.4.4. Implementation details. In order to obtain a detail 3D shape of the object, we evaluate 40 scales. At each scale, the voting space is discretized into bins of 5 pixels by 5 pixels. The aspect ratio of the object is also descritized into about 10 bins (depending on the object category). In order to achieve high maximum recall, we allow the detector to return as many as 1000 candidates with scores higher than 0.01. After non-maximum suppression, we

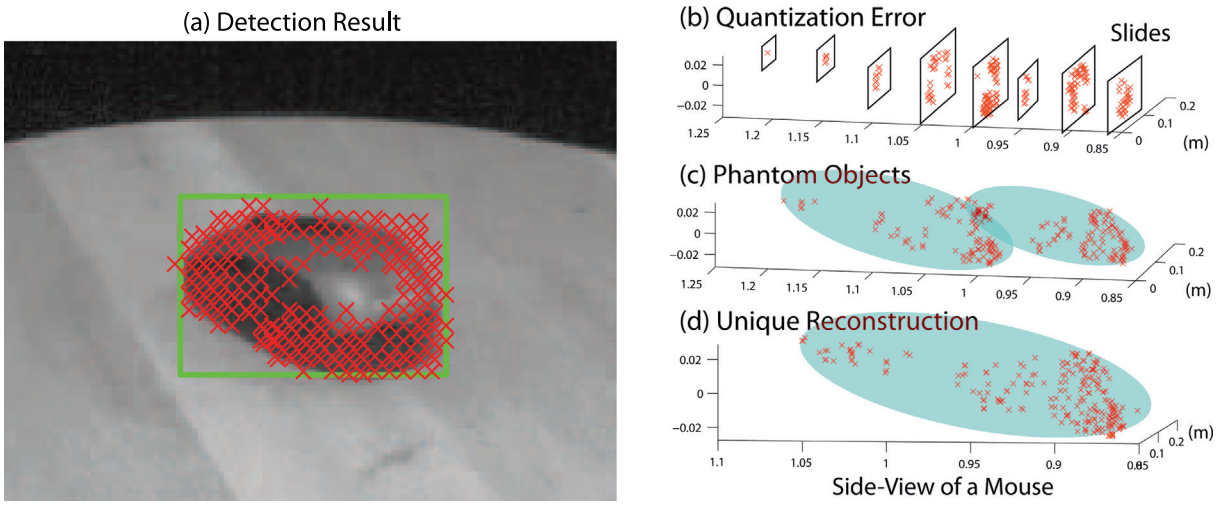

Fig. 7. A typical detection result in (a) shows object hypothesis bounding box (green box) and patches (red crosses) vote for the hypothesis. A naive reconstruction suffers from quantization error (b) and phantom objects (c). Our algorithm overcomes these issues and obtains (d). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

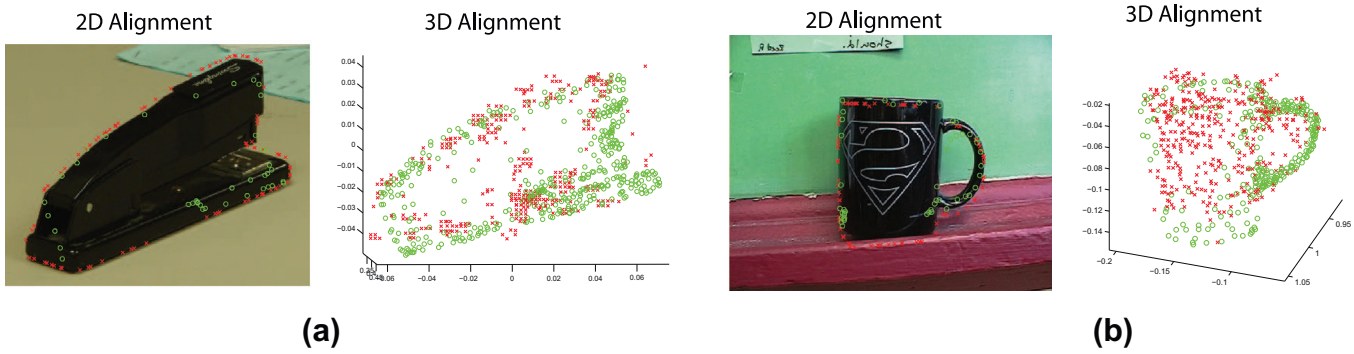

Fig. 8. Two examples of 3D + 2D ICP fitting. In (a and b) (Left), the 2D contour alignment results are shown, where a subset of points on the 2D object contour are indicated by red crosses, and projected vertices lying on the occluding boundary of the 3D CAD model are indicated by green dots. In (a and b) (Right), the 3D points alignment results are shown, where the partial/sparse inferred point clouds (by DEHV) are indicated by red crosses, and the vertices of the 3D CAD model are indicated by green dots. Notice that these two alignments are jointly enforced by Eq. (6). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 9. Hole filling results using (Left) classic Poisson compositing, and (Right) our error-tolerant compositing technique. Notice that red circles highlight regions where the bleeding artifact is fixed by the error-tolerant technique. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

typically obtain less than 100 candidates, and the correct detections are typically among the top few candidates. Training DEHV for each class with one tree takes less than one hour and trees can be trained in parallel. Detection using DEHV takes less than one minute per image. The 2D + 3D ICP is completed less than a second for each CAD model, and the total time for 2D + 3D ICP is linearly proportional to the number of CAD models. The texture completion step is done in a few seconds. Therefore, the overall process will take about a few minutes in our experiment using less than 10 CAD models.

3.2. Stage 2: 3D modelling

The goal of 3D modelling is to obtain the full 3D shape and texture of an (unknown) object from a single images portraying the object observed from an (unknown) viewpoint. We can achieve this by using the inferred depths from the image (Section 3.1), which is a partial (view point limited) 3D point cloud (Partial Shape) of the object (Fig. 1c). Here we discuss details on how to complete the partial reconstruction.

3.3. 3D shape recovery

We adopt the idea of using 3D shape exemplars to help recover the missing portions of object 3D surface. The idea (similar to [65]) is to find a 3D shape exemplar from a given database of 3D shape that can be aligned to the existing incomplete 3D structure. As a result of this alignment, the incomplete elements of the surface can

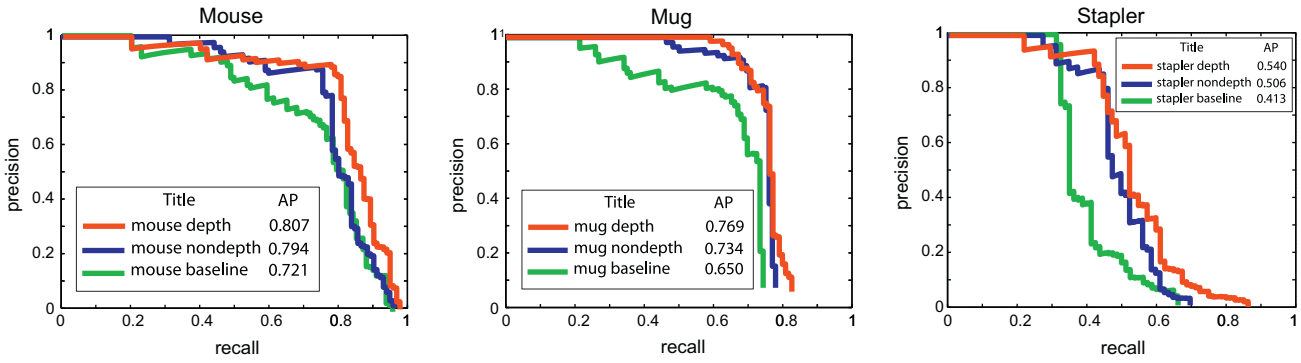

Fig. 10. Object localization results are shown as precision recall curves evaluated using PASCAL VOC protocol. (Green curve) Result using standard ISM model (baseline). (Blue curve) Result using DEHV with no depth information during testing. (Red curve) Result using DEHV with partial depth information during testing. Notice the consistent improvement of average precision (AP) compared to the baseline hough voting. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

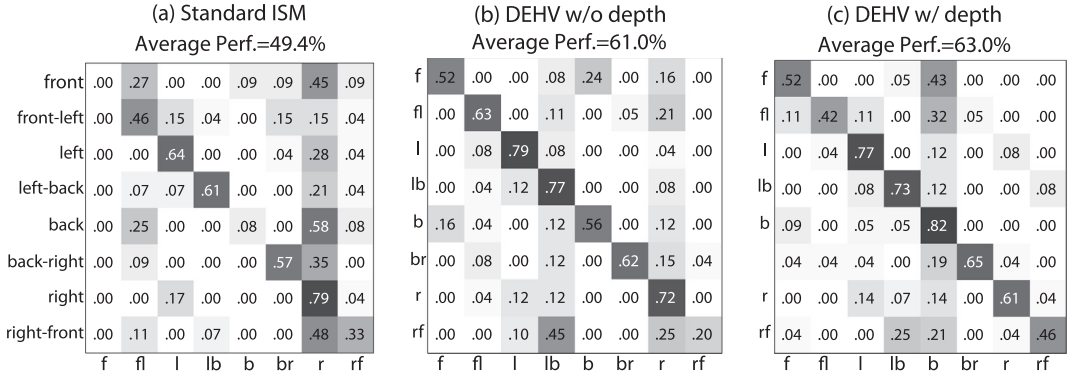

Fig. 11. Pose estimation results averaged across three categories. The average accuracy increases when more 3D information is available. Notice that, when depth is available in both training and testing, the best performances are achieved.

be filled (replaced) with those of the aligned 3D exemplar. The challenges are: (i) how to search efficiently in the database of 3D shape exemplars until the most suitable shape is found. (ii) perform accurate alignment so as to enable accurate replacement. The first challenge is addressed by leveraging the DEHV detector's ability to return object class and pose labels. This greatly reduces the search space and allows to extract from the dataset a subset of exemplars that are likely to be very similar to the one we seek to reconstruct.

We carry out accurate alignment between the reconstructed 3D shape and the exemplar 3D shape using a novel ICP algorithm. This novel ICP performs alignment jointly in 3D shape as well as in image space. The alignment in 3D shape is carried out between vertices of a 3D exemplar model and the reconstructed 3D points. The alignment in image space is carried out between the projected occluding boundaries of the 3D exemplar model and object 2D contour. In the image, 2D contours are obtained by applying grabcut foreground segmentation algorithm [66] within the detection window. This joint alignment process is obtained by minimizing the following cost function,

$$C(T) = \sum_i C_3(q_i, T(v_i)) + \lambda \sum_j C_2(e_j, Proj(T(v_{o_j}))) \quad (6)$$

The first term, $C_3(q_i, T(v_i))$ evaluates the 3D distance between an inferred 3D point q_i and the transformed corresponding vertex $T(v_i)$, where $T(\cdot)$ applies a 3D affine transform on a vertex v_i. The second term,

$$C_2(e_j, Proj(T(v_{o_j}))),$$

Table 1
Depth error comparison between our method and the baseline method. Notice that our errors are always lower than the baseline errors.

	Abs. depth in (m) (known focal length) Sparse/baseline	Rel. depth (unknown focal length) Sparse/baseline
Mouse	0.0145/0.0255	0.0173/0.0308
Mug	0.0176/0.0228	0.0201/0.0263
Stapler	0.0094/0.0240	0.0114/0.0298

evaluates the 2D distance between a pixel at the object's 2D contour e_j and the 2D projection of the transformed corresponding vertex at the occlusion boundary $(Proj(T(v_{o_j})))$. The parameter λ strikes the balance between two terms and it is chosen empirically. Since the ground truth 3D and 2D correspondences are unknown, the ICP algorithm alternates between 1) finding the transformation T which minimizes the cost $C(T)$ and 2) finding the correspondences which are the closest 3D point $T(v_i)$ to q_i and the closest 2D point $Proj(T(v_{o_j}))$ to e_j, till convergence. By choosing the model corresponding to the smallest cost, we automatically complete the 3D shape which best represents the query object in both 2D and 3D (See Fig. 8). Notice that both terms in Eq. (6) are critical for achieving robust alignment. For instance, the alignment of projected 3D CAD model with the 2D object contour (second term of Eq. (6)) can give rise to erroneous solutions that can be easily fixed if the first term of Eq. (6) is also considered. On the other hand, second term of Eq. (6) is useful to fix small registration errors in 3D which may correspond to large retrojection errors.

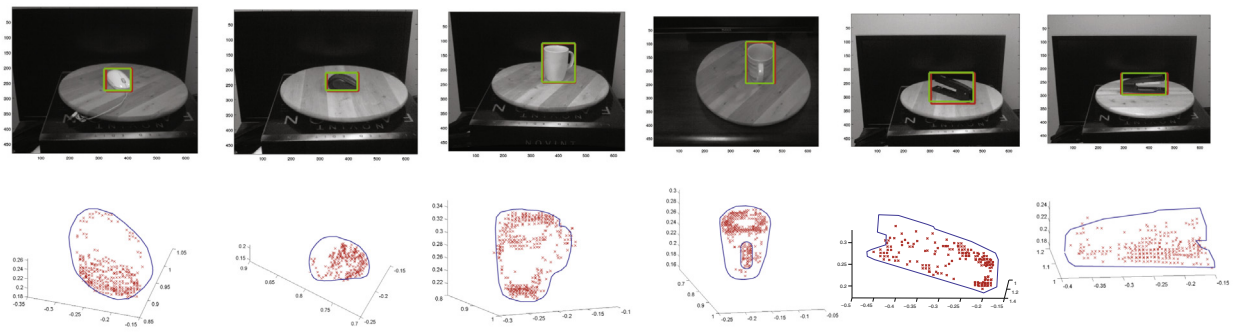

Fig. 12. Example of object detections (Top) and inferred 3D point clouds (Bottom). The inferred point clouds preserve the detailed structure of the objects, like the handle of mug. Object contours are overlaid on top of the image to improve the readers understanding. Please refer to the author's project page for a better visualization.

(a)

Hough Detector recall @ 1.0 FPPI			
DEHV	ISM[1]	M^2HT[9]	PMKrank [10]
87.1	35.5	55.0	74.2

Non-Hough Detector recall @ 0.3/0.4 FPPI	
DEHV	KAS [3]
77.4/80.6	67.8/77.4

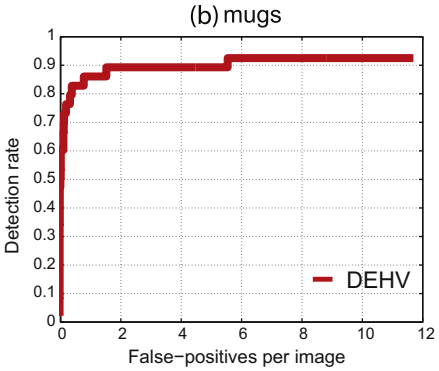

Fig. 13. Performance on the mug category of ETHZ shape dataset [3]. (a-Top) Performance comparison with other pure Hough voting methods (M^2HT) [9] and (PMK rank) [10]. (a-Bottom) Performance comparison between state-of-the-art non-hough voting methods [3]. (b) Detection rate vs. FPPI of DEHV.

Table 2
Pose estimation performance on 3D object dataset [21].

DEHV stapler	DEHV mouse	Savarese et al. 2008 [4]	Farhadi et al. 2009 [73]
75.0	73.5	64.78	78.16

3.4. Texture completion

After shape alignment (Fig. 1d), we can directly map the texture from the image inside the 2D object contour onto the 3D model. This simple approach gives us a model with incomplete texture (see Fig. 1e), where occluded object regions will not be assigned to any texture. In order to obtain a model with complete texture, we propose the following two approaches to infer the texture of the occluded regions of the 3D model.

3.4.1. Symmetric property

We use the property that object categories have often symmetric topology to transfer the texture from the visible regions to the invisible ones (see Fig. 1f). Specifically, we assume that the object shape of the categories of interest are approximately bilateral symmetric (that is, they are symmetric with respect to a plane of reflection). Most common man-made objects satisfy this property. The identification of the bilateral symmetry is carried automatically by applying the symmetry detection algorithm by [67] to the registered CAD model. This algorithm allows to detect the plane of reflection. After the plane of reflection is detected, we identify the pairs of faces which are in symmetric correspondence across the plane of reflection. By knowing symmetric pairs of faces, we transfer the texture from the visible surface areas (group of faces) to the invisible ones as follows: (i) Since faces are either on the left or right side of the plane of reflection, we decide which group (left or right of the plane of reflection) are most visible. The texture coordinates of the vertices composing the faces in the less visible group are removed. (ii) The remaining texture coordinates are transferred to their symmetric correspondences.

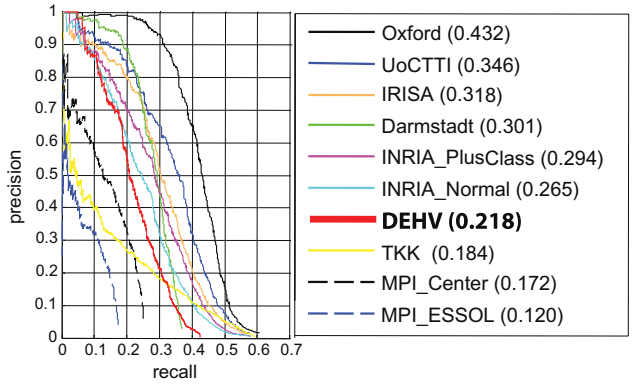

Fig. 14. Object localization result using PASCAL VOC07 dataset. The precision-recall generated by our method (red) is compared with the results of 2007 challenge [2]-Oxford, [2]-UoCTTI, [2]-IRISA, [2]-Darmstadt, [2]-INRIAPlusClass, [2]-INRIANormal, [2]-TKK, [2]-MPICenter, [2]-MPIESSOL. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

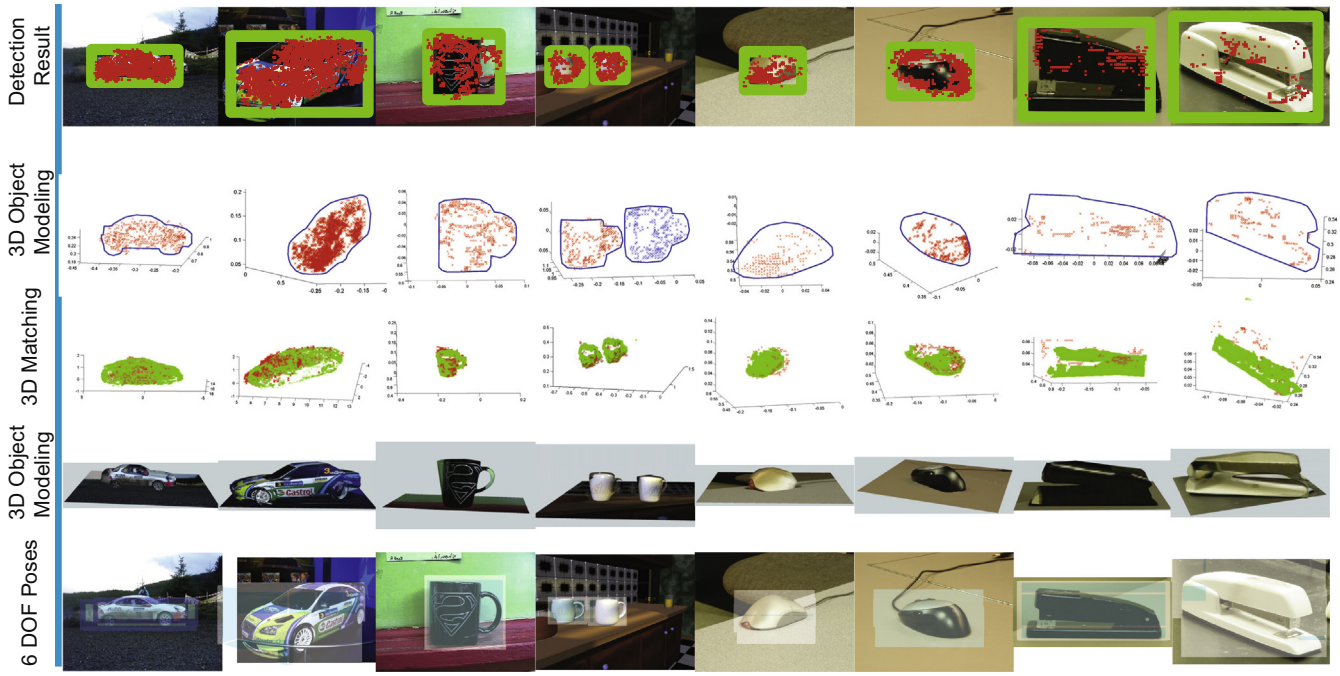

Fig. 15. Examples of the complete 3D object inference process using the testing images from Pascal VOC07 [2], ETHZ Shape [3], and 3D object dataset [13]. This figure should be viewed in colour. Row 1 Detection results (green box) overlaid with image patch centers (red cross) which cast the votes. Row 2 Inferred 3D point clouds (red dots), given the detection results. Row 3 3D registration results, where red indicates the inferred partial point clouds and green indicates the visible parts of the 3D CAD model. Row 4 3D Object modelling using the 3D CAD models and estimated 3D pose of the objects. Notice that the supporting plane in 3D object modelling are manually added. Row 5 Visualizations of the estimated 6 DOF poses. (See author's project page for 3D visualization.) (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

	Relative Depth Error Dense/Baseline				
	Mouse	Mug	Stapler	Car	Bicycle
3D ICP	0.0140/0.0216	0.0287/0.0252	0.0271/0.0283	0.0770/0.1038	0.0630/0.0631
2D+3D ICP	0.0113/0.0209	0.0227/0.0295	0.0260/0.0360	0.0900/0.1189	0.0563/0.0607

Fig. 16. This table shows the median of the relative depth errors for inferred depths obtained after both just 3D ICP (Top-Row) and joint 2D + 3D ICP (Bottom-Row) CAD model alignment. Notice that relative depth error is defined as $\frac{\|d-\hat{d}\|}{d}$, where d is the ground truth depth, and $\hat{d}$ is the estimated depth. Notice that $\hat{d}$s for each object instance are scaled so that ds and $\hat{d}$s have the same median so that inconsistent differences between median depths will not influence the evaluation of 3D shape reconstruction.

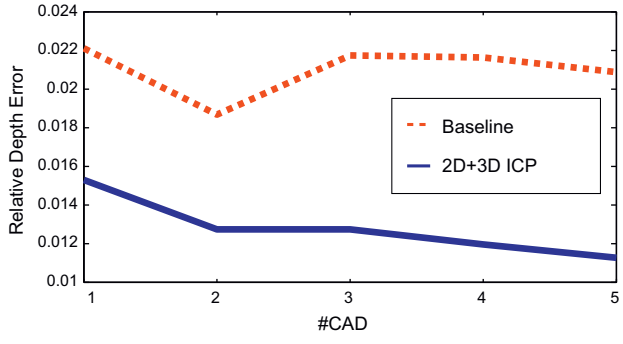

Fig. 17. Relative depth errors using different number of CAD models for 2D + 3D ICP.

3.4.2. Hole filling

The property of symmetry discussed above does not guarantee that all surface elements are filled or assigned to object texture. Typically, the resulting models will still have small holes on the surface (see Fig. 1f). A rich line of work [68–71] have studied the problem of image completion or hole filling only on the 2D domain. In this paper, we apply an error-tolerant image compositing technique (inspired by [6]) to the un-textured region (holes in Fig. 1f). Instead of solving the classic poisson equation [72], we solve the following weighted equation:

$$div(W(\nabla I - v)) = 0 \tag{7}$$

where I is the unknown image, v is the gradient field to guide the texture completion process, and W is the weight capturing the importance of the gradient field. W is introduced in [6] so that the error between the image ∇I and the gradient field v is not evenly distributed which causes the typical bleeding artifacts (Fig. 9). In our implementation, we extract the boundary RGB value from the image and simply assume a uniform gradient field v within region (hole). Most importantly, we set W such that all interior pixels correspond to a constant weight, except for pixels lying on the edges between pairs of faces with very different surface normals corresponds to zero weights. The weights corresponding to boundary pixels are set such that if a boundary colour is very different from the median colour of its neighboring boundary pixels, its corresponding weight is low, and vice versa. In order to fill all the holes, we first group the faces without texture to a set of disjoint groups, where faces in different groups do not share vertices. For each group, we find the hole boundary which shares vertices with the faces with texture, and extract the RGB value from the faces

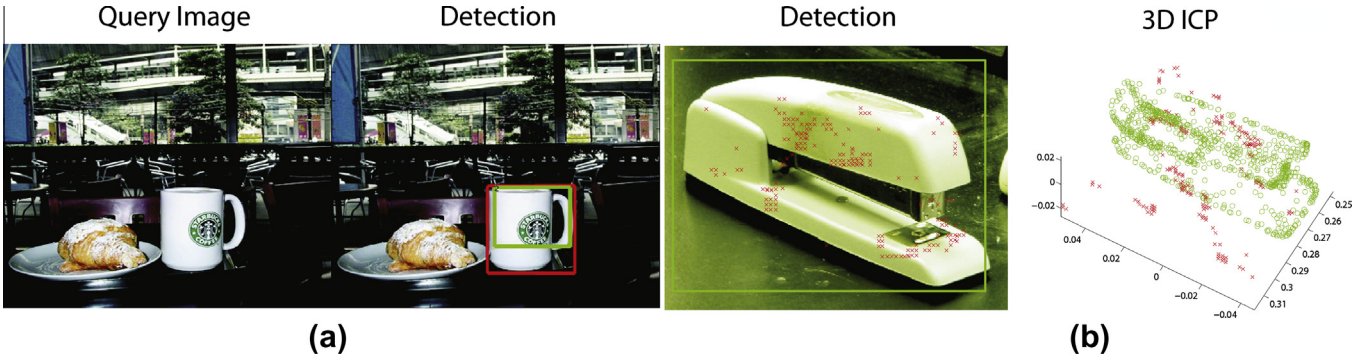

Fig. 18. Examples of typical failures. (a): Ground truth object bounding box and candidate object bounding box are shown in red and green respectively. In this case, our DEHV detector fails to locate the object in the image. Hence, the following steps to reconstruct the object will be poorly performed. (b): The object is detected correctly. However, the 3D ICP algorithm fails to align the CAD model (green) to the inferred partial/sparse point clouds (red), since the inferred point clouds are too sparse. In this case, the object 2D contour information is very uesful for improving the alignment result. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

with texture along the hole boundary. We then project the group of faces without texture onto an image plane with a most frontal view and solve I in Eq. (7) to fill the image RGB value within the projected hole boundary.

4. Experiment

We conduct experiments to evaluate the object detection and shape recovery performance of our DEHV algorithm in Section 4.1, and the quality of 3D modelling in terms of both shape recovery and texture completion in Section 4.2. Typical failure cases of the object detector and 3D ICP are shown in Fig. 18(a,b) respectively.

4.1. Evaluation of DEHV

We evaluated our DEHV algorithm on several datasets: ETHZ Shape dataset [3], 3D object dataset [13], and Pascal VOC07 dataset [2]. The training settings were as follows. For each training image, we randomly sample 100 image patches from object instances and 500 image patches from background regions. The scale of the patch size from the corresponding object instance is determined by its (known) depth (Fig. 6). At the end, 10 random trees (Section 3.1.3) are trained using the sampled foreground and background patches for each dataset. For each experiment, we use a Hog-like feature introduced in [8]. During detection, our method treats each discrete viewpoint as a different class O.

4.1.1. Exp.I: System analysis on a novel 3D table-top object dataset

Due to the lack of datasets comprising both images and 3D depth maps of set of generic object categories, we propose a new 3D table-top object category dataset collected on a robot platform. The dataset contains three common table-top object categories: mice, mugs, and staplers, each with 10 object instances. We arrange these objects in two different sets for the purpose of object localization and pose estimation evaluation. The object localization dataset (Table-Top-Local) contains 200 images with the number of object ranging from 2 to 6 object instances per image in a clutter office environment. The object pose estimation dataset (Table-Top-Pose) contains 480 images where each object instance is captured under 16 different poses (eight angles and two heights). For both settings, each image comes with depth information collected using a structure-light stereo camera. Please see the author's project page (http://www.eecs.umich.edu/sunmin) for more information about the dataset.

We evaluate our method under three different training and testing conditions, which are (1) standard ISM model trained and tested without depth, (2) DEHV trained with depth but tested without depth, and (3) DEHV trained and tested with depth. We show that the knowledge of 3D information helps in terms of object localization (Fig. 10), and pose estimation (Fig. 11). Moreover, we evaluate our method's ability to infer depth from just a single 2D image. Given the ground truth focal length of the camera, we evaluate the absolute depth error for the inferred partial point clouds in Table 1-Left Column. Notice that our errors are always lower than the baseline errors.[1] We also evaluate the relative depth errors[2] reported in Table 1-Right Column when the exact focal length is unknown. Object detection examples and inferred 3D point clouds are shown in Fig. 12.

4.1.2. Exp.II:Comparision on three challenging datasets

In order to demonstrate that DEHV generalizes well on other publicly available datasets, we compare our results with state-of-the-art object detectors on a subset of object categories from the ETHZ shape dataset, 3D object dataset, and Pascal 2007 dataset. Notice that all of these datasets contain 2D images only. Therefore, training of DEHV is performed using the 2D images from these public available dataset and the depth maps available from the 3D table-top dataset and our own set of 3D reconstruction of cars[3].

4.1.2.1. ETHZ shape dataset. We test our method on the Mug category of the ETHZ Shape dataset. It contains 48 positive images with mugs and 207 negative images with a mixture of apple logos, bottles, giraffes, mugs, and swans. Following the experiment setup in [3], we use 24 positive images and an equal number of negative images for training. We further match the 24 mugs with the mugs in 3D table-top object dataset to transfer the depth maps to the matched object instances so that we obtain augmented depth for positive training images. All the remaining 207 images in the ETHZ Shape dataset are used for testing.

The table in Fig. 13a-top shows the comparison of our method with the standard ISM and two state-of-the-art pure voting-based methods at 1.0 False-Positive-Per-Image (FPPI). Our DEHV method (recall 83.0 at 1 FPPI) significantly outperforms Max-Margin Hough Voting (M^2HT) [9] (recall 55 at 1 FPPI) and pyramid match kernel

[1] It is computed assuming each depth is equal to the median of the depths of the inferred partial point clouds.

[2] $\frac{\|d-\hat{d}\|}{d}$ where d is the ground truth depth, and $\hat{d}$ is the estimated depth. $\hat{d}$ is scaled so that d and $\hat{d}$ have the same median.

[3] Notice that our own dataset is only used to provide depth information.

Fig. 19. Examples of semi-automatic 3D object modelling process on a number of query images. This figure is best viewed in colour. Col. (a) Sample detection results (green bounding box). Col. (b) Partial/Sparse reconstruction of the detected object, where the inferred point clouds in red. Col. (c) Incomplete object 3D models using only the visible part of the registered 3D CAD model. Col. (d) Complete 3D model after texture completion using symmetric properties and hole filling. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

ranking (PMK ranking) [10] (recall 74.2 at 1 FPPI). The table in Fig. 13a-bottom shows that our method is superior than state-of-the-art non-voting-based method KAS [3]. Note that these results are not including a second stage verification step which would naturally boost up performance. The recall vs (FPPI) curve of our method is shown in Fig. 13b.

4.1.2.2. 3D object dataset. We test our method on the mouse and stapler categories of the 3D object dataset [13,4], where each category contains 10 object instances observed under eight angles, three heights, and two scales. We adapt the same experimental settings as [13,4] with additional depth information from the first five instances of the 3D table-top object dataset to train our DEHV

models. The pose estimation performance of our method is shown in Table 2. It is superior than [4] and comparable to [73] (which primarily focuses on pose estimation only).

4.1.2.3. Pascal VOC 2007 dataset. We tested our method on the car category of the Pascal VOC 2007 challenge dataset [2], and report the localization performance. Unfortunately PASCAL does not contain depth maps. Thus, in order to train DEHV with 3D information, we collect a 3D car dataset containing five car instances observed from eight viewpoints, and use Bundler [74] to obtain its 3D reconstruction. We match 254 car instances [4] in the training set of Pascal 2007 dataset to the instances in 3D car dataset and associate depth maps to these 254 Pascal training images. This way the 254 positive images can be associated to a rough depth value. Finally, both 254 positive Pascal training images and the remaining 4250 negative images are used to train our DEHV detector. We obtain reasonably good detection performance (Average Precision 0.218) even though we trained with fewer positive images (Fig. 14). Detection examples and inferred objects 3D shape are shown in Fig. 15.

4.2. Evaluation of 3D modelling

We conduct experiments to evaluate quantitatively and qualitatively the 3D modelling stage of our system (Stage 2 Section 3.2). At that end, we collect a dataset which comprises 3D reconstructions of five object categories: mice, mugs, staplers, cars, and bicycles. For each category, the dataset includes about three object instances and each instance contains images of the object from camera poses evenly sampled across multiple azimuth angles. The corresponding depth information of each image is either collected from a structured-light stereo camera or a structure from motion method.

We evaluate our method's ability to recover the full 3D shape from an inferred rough 3D structure (output of stage 1). Relative depth errors between ground truth depths and recovered depths (i.e. these obtained after both just 3D ICP (Top-Row) and joint 2D + 3D ICP (Bottom-Row) CAD model alignment) are shown in Fig. 16. Baseline errors are computed assuming the depths are all equal to the median of the inferred depths. Notice that the errors of 2D + 3D ICP are always smaller than the baseline errors, and the errors of 2D + 3D ICP are always smaller or similar than the errors of 3D ICP. In our experiments, the inferred 3D and 2D information are matched with about 5 different 3D CAD models selected from the database with the correct object category and pose. The database of 3D CAD models is either collected from [11] and other online 3D warehouses, or obtained by shape from silhouette [12]. Fig. 17 shows a plot of the relative depth errors of 2D + 3D ICP versus the number of CAD models of mouse being used. The plot suggests that the more CAD models are used in 2D + 3D ICP, the smaller the error in registration is.

We have further used the ETHZ Shape mug dataset [3] and 3D object dataset [13] to generate typical examples of 3d reconstructions from a single view. Fig. 19 shows qualitative results of our full algorithm on several images from 3D object dataset, ETHZ Shape mug dataset, 3D table-top object dataset, and 3D modelling dataset.

5. Conclusion

We proposed a new detection scheme called DEHV which can successfully detect objects, estimate their pose from either a single 2D image or a 2D image combined with depth information. Moreover, we demonstrated that DEHV is capable of recover the 3D shape of object categories from just one single uncalibrated image. Given such a partial 3D Shape of the object, we show that novel 3D shape recovery and texture completions techniques can be applied to fully reconstruct the 3D model of the object with both complete shape and texture. As future work, we envision the possibility of integrating more sophisticated texture or 3D shape completion techniques for further improving the quality of the overall 3D model on a large scale of object categories.

[4] 254 cars is a subset of the 1261 positive images in the PASCAL training set. The subset is selected if they are easy to match with the 3D car dataset.

Acknowledgments

We acknowledge the support of NSF (Grant CNS 0931474) and the Gigascale Systems Research Center, one of sixresearch centers funded under the Focus Center Research Program(FCRP), a Semiconductor Research Corporation Entity, Google Research Award(SC347174), and Willow Garage Inc. for collecting the 3D table-top object category dataset.

References

[1] B. Leibe, A. Leonardis, B. Schiele, Combined object categorization and segmentation with an implicit shape model, in: ECCV Workshop on Statistical Learning in Computer Vision, 2004.
[2] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, 2007.
[3] V. Ferrari, L. Fevrier, F. Jurie, C. Schmid, Groups of adjacent contour segments for object detection, IEEE Trans. PAMI 30 (1) (2008) 36–51.
[4] S. Savarese, L. Fei-Fei, View synthesis for recognizing unseen poses of object classes, in: ECCV, 2008.
[5] Microsoft Corp. Redmond WA, Kinect for Xbox 360.
[6] M.K.J. Michael W. Tao, S. Paris, Error-tolerant image compositing, in: ECCV, 2010.
[7] D.H. Ballard, Generalizing the hough transform to detect arbitrary shapes, Pattern Recognition.
[8] J. Gall, V. Lempitsky, Class-specific hough forests for object detection, in: CVPR, 2009.
[9] S. Maji, J. Malik, Object detection using a max-margin hough tranform, in: CVPR, 2009.
[10] B. Ommer, J. Malik, Multi-scale object detection by clustering lines, in: ICCV, 2009.
[11] P. Shilane, P. Min, M. Kazhdan, T. Funkhouser, The princeton shape benchmark, in: Proceedings of the Shape Modeling International 2004, 2004.
[12] A. Laurentini, The visual hull concept for silhouette-based image understanding, IEEE Trans. Pattern Anal. Mach. Intell. 16 (2) (1994) 150–162.
[13] S. Savarese, L. Fei-Fei, 3D generic object categorization, localization and pose estimation, in: ICCV, 2007.
[14] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: CVPR, 2005.
[15] R. Fergus, P. Perona, A. Zisserman, A sparse object category model for efficient learning and exhaustive recognition., in: CVPR, 2005.
[16] H. Schneiderman, T. Kanade, A statistical approach to 3D object detection applied to faces and cars, in: CVPR, 2000.
[17] H. Su, M. Sun, L. Fei-Fei, S. Savarese, Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories, in: ICCV, 2009.
[18] D. Hoeim, C. Rother, J. Winn, 3d layoutcrf for multi-view object class recognition and segmentation, in: CVPR, 2007.
[19] P. Yan, D. Khan, M. Shah, 3d model based object class detection in an arbitrary view., in: ICCV, 2007.
[20] J. Liebelt, C. Schmid, K. Schertler, Viewpoint-independent object class detection using 3d feature maps, in: CVPR, 2008.
[21] M. Arie-Nachimson, R. Basri, Constructing implicit 3d shape models for pose estimation, in: ICCV, 2009.
[22] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, L. Van Gool, Using multi-view recognition and meta-data annotation to guide a robot's attention, Int. J. Rob. Res. 28 (2009) 976–998.
[23] M.R. Oswald, E. Toeppe, K. Kolev, D. Cremers, Non-parametric single view reconstruction of curved objects using convex optimization, in: DAGM, Jena, Germany, 2009.
[24] D.P. Huttenlocher, S. Ullman, Recognizing solid objects by alignment with an image, IJCV 5 (2) (1990) 195–212.
[25] F. Rothganger, S. Lazebnik, C. Schmid, J. Ponce, 3D object modeling and recognition using affine-invariant patches and multi-view spatial constraints., in: CVPR, 2003.
[26] A.C. Romea, D. Berenson, S. Srinivasa, D. Ferguson, Object recognition and full pose registration from a single image for robotic manipulation, in: ICRA, 2009.
[27] D.G. Lowe, Local feature view clustering for 3d object recognition, in: CVPR, 2001.

[28] R.B. Rusu, N. Blodow, Z.C. Marton, M. Beetz, Close-range scene segmentation and reconstruction of 3d point cloud maps for mobile manipulation in human environments, in: IROS, 2009.
[29] T. Deselaers, A. Criminisi, J. Winn, A. Agarwal, Incorporating on-demand stereo for real time recognition, in: CVPR, 2007.
[30] D. Hoeim, S. Savarese, Representations and Techniques for 3D Object Recognition and Scene Interpretation, Morgan and Claypool, 2011.
[31] P.E. Debevec, C.J. Taylor, J. Malik, Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach, in: SIGGRAPH, 1996.
[32] M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, R. Koch, Visual modeling with a hand-held camera, IJCV 59 (3) (2004) 207–232.
[33] N. Snavely, S.M. Seitz, R. Szeliski, Photo tourism: Exploring photo collections in 3d, in: SIGGRAPH, 2006.
[34] S. Agarwal, N. Snavely, I. Simon, S.M. Seitz, R. Szeliski, Building rome in a day, in: ICCV, 2009.
[35] A.R. Dick, P.H.S. Torr, R. Cipolla, Modelling and interpretation of architecture from several images, IJCV 60 (2) (2004) 111–134.
[36] S. Teller, M. Antone, Z. Bodnar, M. Bosse, S. Coorg, M. Jethwa, N. Master, Calibrated, registered images of an extended urban area, IJCV 53 (1) (2003) 93–107.
[37] Y. Horry, K.-I. Anjyo, K. Arai, Tour into the picture: using a spidery mesh interface to make animation from a single image, in: SIGGRAPH, 1997.
[38] D. Liebowitz, A. Criminisi, A. Zisserman, Creating architectural models from images, in: EuroGraphics, 1999.
[39] J. Kopf, B. Neubert, B. Chen, M. Cohen, D. Cohen-Or, O. Deussen, M. Uyttendaele, D. Lischinski, Deep photo: model-based photograph enhancement and viewing, in: SIGGRAPH Asia, 2008.
[40] A. Saxena, M. Sun, A.Y. Ng, Make3d: learning 3d scene structure from a single still image, IEEE TPAMI 31 (5) (2009) 824–840.
[41] D. Hoiem, A.A. Efros, M. Hebert, Automatic photo pop-up, in: SIGGRAPH, 2005.
[42] D.C. Lee, M. Hebert, T. Kanade, Geometric reasoning for single image structure recovery, in: CVPR, 2009.
[43] H. Wang, S. Gould, D. Koller, Discriminative learning with latent variables for cluttered indoor scene understanding, in: ECCV, 2010.
[44] V. Hedau, D. Hoiem, D. Forsyth, Thinking inside the box: Using appearance models and context based on room geometry, in: ECCV, 2010.
[45] A. Schwing, T. Hazan, M. Pollefeys, R. Urtasun, Efficient structured prediction for 3d indoor scene understanding, in: CVPR, 2012.
[46] J.-Y. Bouguet, P. Perona, Visual navigation using a single camera, in: ICCV, 1995.
[47] S. Savarese, M. Andreetto, H. Rushmeier, F. Bernardin, P. Perona, 3d reconstruction by shadow carving: theory and practical evaluation, IJCV 71 (3) (2006) 305–336.
[48] S. Rusinkiewicz, O. Hall-Holt, M. Levoy, Real-time 3d model acquisition, ACM Trans. Graph. 21 (3) (2002) 438–446.
[49] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, D. Fulk, The digital michelangelo project: 3d scanning of large statues, in: SIGGRAPH, 2000.
[50] K.N. Kutulakos, S.M. Seitz, A theory of shape by space carving, IJCV 38 (3) (2000) 199–218.
[51] P.R.S. Mendonça, K.-Y.K. Wong, R. Cipolla, Camera pose estimation and reconstruction from image profiles under circular motion, in: ECCV, 2000.
[52] S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, R. Szeliski, A comparison and evaluation of multi-view stereo reconstruction algorithms, in: CVPR, 2006.
[53] M. Prasad, A.W. Fitzgibbon, A. Zisserman, L.J.V. Gool, Finding nemo: deformable object class modelling using curve matching, in: CVPR, 2010.
[54] L. McMillan, G. Bishop, Plenoptic modeling: an image-based rendering system, in: SIGGRAPH, 1995.
[55] M. Levoy, P. Hanrahan, Light field rendering, in: SIGGRAPH, 1996.
[56] D.G. Aliaga, T. Funkhouser, D. Yanovsky, I. Carlbom, Sea of images: a dense sampling approach for rendering large indoor environments, IEEE Comput. Graph. Appl. 23 (6) (2003) 22–30.
[57] C.L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, R. Szeliski, High-quality video view interpolation using a layered representation, in: SIGGRAPH, 2004.
[58] M. Prasad, A. Fitzgibbon, Single view reconstruction of curved surfaces, in: CVPR, 2006.
[59] C. Colombo, A. Del Bimbo, F. Pernici, Metric 3d reconstruction and texture acquisition of surfaces of revolution from a single uncalibrated view, TPAMI 27 (1) (2005) 99–114.
[60] X. Chen, S.B. Kang, Y.-Q. Xu, J. Dorsey, H.-Y. Shum, Sketching reality: realistic interpretation of architectural designs, ACM Trans. Graph. 27 (2) (2008) 1–15.
[61] O.A. Karpenko, J.F. Hughes, Smoothsketch: 3d free-form shapes from complex sketches, ACM Trans. Graph. 25/3 (2006) 589–598.
[62] N. Jiang, P. Tan, L.-F. Cheong, Symmetric architecture modeling with a single image, in: SIGGRAPH Asia, 2009.
[63] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, L.J.V. Gool, Depth-from-recognition: Inferring meta-data by cognitive feedback, in: 3D Representation for Recognition Workshop in ICCV, 2007.
[64] M. Sun, G. Bradski, B.-X. Xu, S. Savarese, Depth-encoded hough voting for joint object detection and shape recovery, in: ECCV, 2010.
[65] M. Pauly, N.J. Mitra, J. Giesen, M. Gross, L.J. Guibas, Example-based 3d scan completion, in: SGP, 2005.
[66] C. Rother, V. Kolmogorov, A. Blake, "grabcut: interactive foreground extraction using iterated graph cuts, ACM Trans. Graph. 23 (3) (2004) 309–314.
[67] N.J. Mitra, L. Guibas, M. Pauly, Partial and approximate symmetry detection for 3d geometry, ACM Trans. Graph. 25 (3) (2006) 560–568.
[68] P.P. Antonio Criminisi, K. Toyama, Object removal by exemplar-based inpainting, in: CVPR, 2003.
[69] A. Shamir, S. Avidan, Seam carving for media retargeting, Commun. ACM 52 (1) (2009) 77–85.
[70] J. Hays, A.A. Efros, Scene completion using millions of photographs, in: SIGGRAPH, ACM, 2007.
[71] A.A. Efros, T.K. Leung, Texture synthesis by non-parametric sampling, in: ICCV, 1999.
[72] P. Pérez, M. Gangnet, A. Blake, Poisson image editing, ACM Trans. Graph. 22 (3) (2003) 313–318.
[73] A. Farhadi, M.K. Tabrizi, I. Endres, D. Forsyth, A latent model of discriminative aspect, in: ICCV, 2009.
[74] N. Snavely, S.M. Seitz, R. Szeliski, Photo tourism: exploring photo collections in 3d, in: SIGGRAPH, 2006.