IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 16, 2020, accepted February 10, 2020, date of publication February 19, 2020, date of current version February 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2975106

Joint Image and 3D Shape Part Representation in Large Collections for Object Blending

ADRIAN PENATE-SANCHEZ AND LOURDES AGAPITO
Department of Computer Science, University College London, London WC1E 6BT, U.K.

Corresponding author: Adrian Penate-Sanchez (adrian@robots.ox.ac.uk)

This work was supported by the SecondHands Project, funded from the EU Horizon 2020 Research and Innovation Programme under Grant 643950.

ABSTRACT We propose a new approach to perform object shape retrieval from images, it can handle the shape of the part of the object and combine parts from different sources to find a different 3D shape. Our method creates a common representation for images and 3D models that enables mixing elements from both kinds of inputs. Our approach automatically extracts the desired part and its 3D shape from each source without the need of annotations. There are many applications to combining parts from images and 3D models, for example, performing smart online catalogue searches by selecting the parts that we are looking for from images or 3D models and retrieve a 3D shape that has the desired arrangement of parts. Our approach is capable of obtaining the shape of the parts of an object from an image in the wild, independently of the pose of the object and without the need of annotations of any kind.

INDEX TERMS Shape blending, joint image and shape embedding, 3D shape, computer vision, computer graphics.

I. INTRODUCTION

The widespread availability of low cost high quality cameras and 3D sensing devices has recently enabled the computer vision and graphics communities to collect and curate vast Internet collections of images and 3D shapes of everyday objects such as *ImageNet* or *ShapeNet*. These datasets have quickly become the cornerstone of tasks such as visual recognition and 3D scene understanding and have led to huge progress since they represent the labelled examples from which machines can reason about shape and appearance.

As these databases of images and 3D shapes keep growing in size and number, organizing and exploring them has become increasingly complex. While most tools developed so far have dealt with shape and appearance modalities separately, some recent methods [2], [3] have begun to exploit the complementary nature of these two sources of information and to reap the benefits of creating a common representation for images and 3D models. As images and 3D shapes are linked together, many possibilities open up to transfer what is learnt from one modality to another. Creating a joint embedding for images and 3D models allows to retrieve 3D models based on image queries (or vice-versa) or to align images of similar 3D shapes – more generally it facilitates the comparison between objects represented in either modality.

However, recent retrieval methods still fall short of being flexible enough to allow advanced queries. Crucially, they are limited to reasoning about objects as a whole – taking a single query image (or shape) as input at test time prevents them from combining object properties from different inputs.

We present **3D Pick 'n' Mix,** a new shape retrieval system that overcomes this limitation by introducing the ability to reason about objects at the level of their constituent parts. Our new approach can formulate more advanced and semantically meaningful search queries such as: *"find me the 3D model that best combines the design of the backrest of the chair in image* 1 *with the shape of the legs of the chair in image* 2*"* (as depicted in Fig. 1) or *"retrieve chairs with wheels"*.

Contributions: The three main contributions of our 3D Pick 'n' Mix system are:

- We learn embeddings for object parts (for instance the legs or the armrests of chairs) which allow us to retrieve images or 3D models of objects with similarly shaped parts.
- We propose a new deep architecture that can map RGB images onto these embeddings of parts by regressing

The associate editor coordinating the review of this manuscript and approving it for publication was Jinjia Zhou.

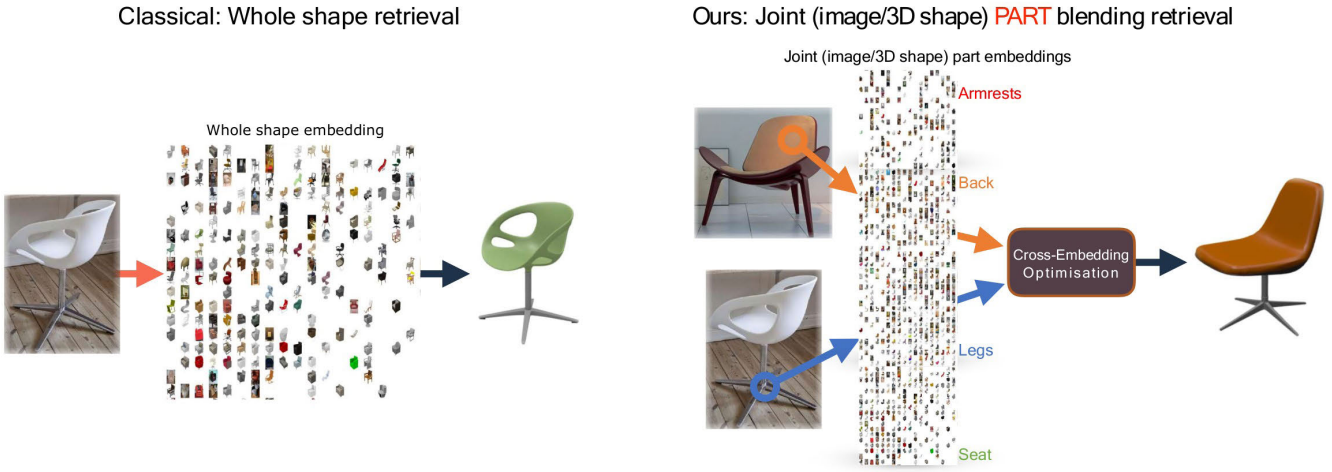

FIGURE 1. The proposed approach can take 2 inputs, either an RGB image or a 3*D* model, identify the parts that compose the object and find a matching shape from Shapenet [1]. Its important to underline that the approach does not need the part to be labelled or segmented in the image. It can also merge parts independently of the type of input source, as seen in the figure.

their coordinates. Crucially, the input to the network is simply an RGB image and the name (label) of the object part. The CNN must therefore learn first to segment the pixels that depict the chosen object part and then to regress its coordinates on the according shape embedding.

- At query time our retrieval system can combine object parts or properties from multiple input images, enabled by a cross-embedding optimization technique.

Our system takes the concept of a joint representation for images and shapes one step further to allow the ability to reason about objects at the level of their constituent parts. 3D Pick 'n' Mix can take more than one image as input, can pick a different object property/part from each image and retrieve a single shape that blends the different object properties/parts together.

This ability to reason at the level of parts provides users with a new level of freedom to explore 3D shape and image datasets. Users could browse through catalogues of furniture and **pick** and **mix** parts, combining for example the legs of a favorite chair from one catalogue and the armrests from another. An example of this kind is shown in Fig. 1.

The **training** of our system requires two steps:

- First we build independent embedding shape spaces for each object part. Embeddings are learnt using the Light-field descriptor [4] of each part which allows a common representation for images and 3D models. Fig. 4 describes the building of shape embeddings.
- Secondly, a CNN embedding coordinate regressor is trained to map real images of an object to the part embeddings. Our novel deep learning architecture jointly performs semantic segmentation of object parts and learns to regress the coordinates of each part on the corresponding part shape embedding. This network is trained using only synthetic data. Fig. 2 illustrates the architecture.

At **test** time the user provides two (or more) images (or 3D models) as input and determines which parts they would like to pick from each (note that this only requires a name such as *'legs'*). Our system retrieves the model that best fits the arrangement of parts by performing a cross-embedding optimization (see Fig. 1). In this work we seek to consolidate and expand into a journal the work already presented in the Asian Conference on Computer Vision that is available in Arxiv [5].

This Journal paper aims to be the definitive publication on this line of work. For this reason we have created a much more detailed and rich manuscript that includes more qualitative and quantitative analysis and made public the code and the datasets. These are the improvements with respect to the conference paper:

- Better description of the method and the insights. The whole paper has been extended from 6.7 thousand words to more than 10 thousand words. This is due to: 1. A deeper and more detailed discussion of the method and the design of the different parts involved in the part shape retrieval, 2. A more thorough review of the related work, 3. Discussion of several additional experimental results to better show the performance of the method, 4. Discussion of incorrect results to show how the method performs in its failure cases. 5. introduction of better and more figures that help to understand and describe the work.
- Experiment depicted in Fig. 12. This additional result shows how the method performs when trying to combine the different possible part arrangements. To obtain this detailed results the ExactPartMatch dataset had to be further annotated to be capable of quantifying the results depending on the part in question.
- Experiment depicted in Fig. 15. This experiment involved taking partly occluded images of Ikea chairs from the internet, annotating them and testing our

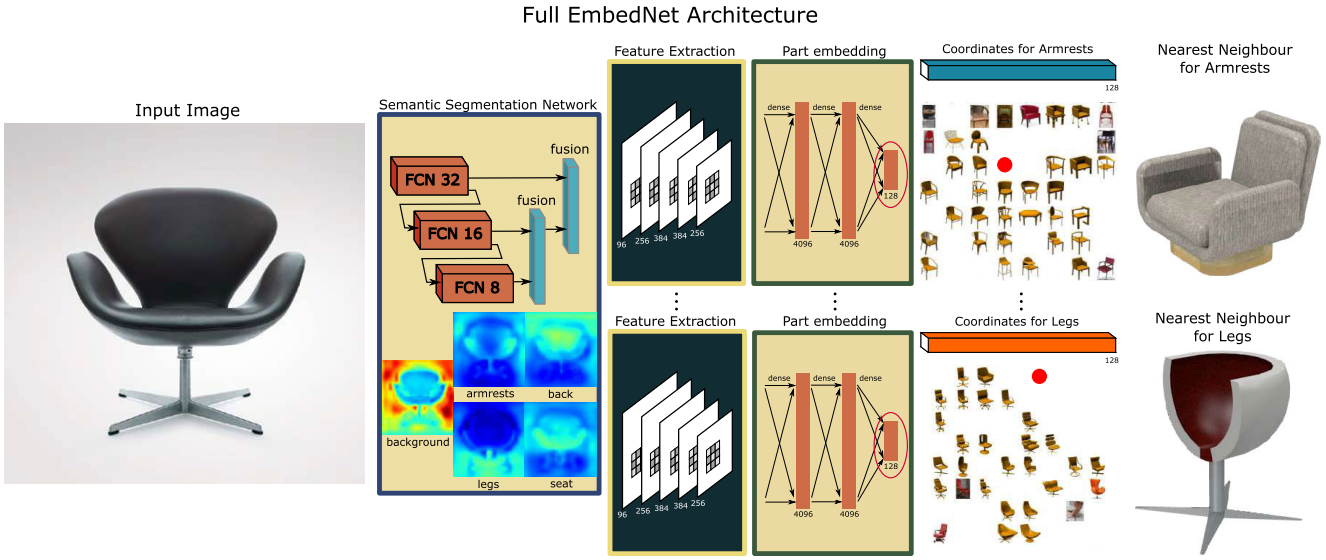

FIGURE 2. Summary of the architecture of EmbedNet, our new deep network that takes an image as input and learns to regress the coordinates of each object part in the different part embeddings. The architecture has 3 sections: the first set of layers performs semantic segmentation of the image pixels into different semantic parts (such as *"backrest", "seat", "armrests" or "legs"* in the case of chairs). The second section learns an intermediate feature representation for embedding coordinate regression. The final section learns to regress the shape coordinates in each of the part embeddings. We show the nearest neighbour shapes found on the *"armrests"* and *"legs"* embeddings for the depicted input image. The results show that objects with similarly shaped parts are found in both cases.

method against the state of the art in whole shape retrieval to show how by understanding the parts separately a whole shape restrieval could be managed when whole shape methods fail.

- Experiment depicted in Fig. 16. This figure includes erroneous examples of shape estimation of our method. We provide them to further give insights of how our method performs.

II. RELATED WORK

Our work overlaps with several well defined lines of research. In this section we will outline recent work on: *(i)* learning joint embeddings for 3D models and images; *(ii)* shape blending and mixing; and *(iii)* modeling parts of 3D models. Other recent papers that are relevant to this work can be found in [6]–[8].

A. JOINT 3D MODEL/IMAGE EMBEDDINGS

While most shape retrieval methods had traditionally dealt with shape and appearance modalities separately, a recent trend has emerged that exploits the complementary nature of appearance and shape by creating a common representation for images and 3D models. Reference [3] exploits the different advantages of shape and images by using the robustness of 3D models for alignment and pose estimation and the reliability of image labels to identify the objects. While they do not explicitly create a joint embedding based on shape similarity they do rely on image representations for both modalities. Our approach is perhaps most closely related to [2] who first build a low dimensional representation of 3D shapes by using the Light Field descriptor (**LfD**) [4] followed by a deep learning approach to map images onto the embedding. However, unlike our approach, their representation is limited to objects as a whole preventing the combination of properties taken from different inputs.

Reference [9] perform shape retrieval from sketches, words, depth maps and real images by creating a embedding space that combines the different inputs. Since intra-class similarity is not the main focus, most instances of the same class tend to appear clustered. Reference [10] learn a embedding-space metric by using triplets of shapes where the first is similar to the third but dissimilar to the second. A deep network is then trained to pull together shapes that are similar while pushing away dissimilar ones. Similarly to our approach, the metric space is defined based on shape and not image similarity. Reference [11] first generate voxel representations of the objects present in the RGB image inputs. A shared latent shape representation is then learnt for both images and the voxelized data. At test time RGB convolutions and volume generation deconvolution layers are used to produce the 3D shape.

Notably, while all the above approaches find common representations for images and 3D shapes and can combine the use of both modalities, unlike us they are restricted to reasoning about object instances and cannot blend together parts or properties from different query images.

B. 3D SHAPE BLENDING/MIXING

Much in the line of the work presented in this paper there has been fruitful research in shape blending in recent years. Reference [12] use parts obtained from several 3D models to

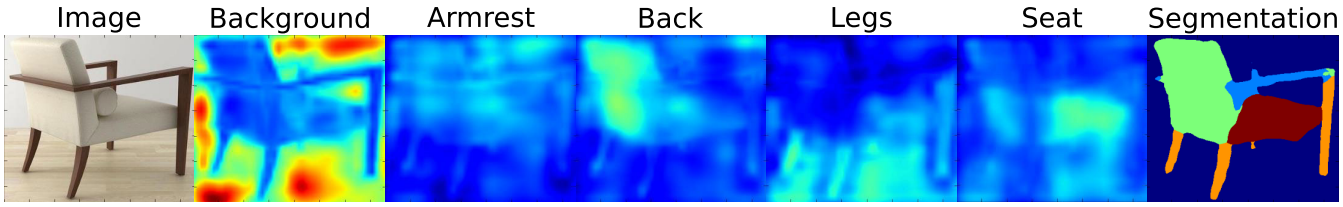

FIGURE 3. Example of the semantic segmentation performed by the first stages of our architecture. We can see the output probabilities for each of the parts and the background give a very strong prior of were the parts of an object can be found. This makes the task of predicting the shape of each of the parts, by estimating their coordinates in the shape embedding, a much easier task. Not requiring labels for each part in the input image makes our approach very easy to use and increases dramatically its applicability.

compose a new 3D model. Unlike our approach, they focus on parts that are defined by their geometry rather than by their semantics. The *"3D model evolution"* approach of [13] takes a small set of 3D models as input to generate many. Parts from two models cross-over to form a new 3D model, continuing to merge original models with new ones to generate a large number of 3D models. Reference [14] generate new shapes by interpolating and varying the topology between two 3D models. The topology is defined via a connectivity graph that is modified to fit the changes in topology in the new blended shape. The photo-inspired 3D modeling method of [15] takes a single image as input, segments it into parts using an interactive model-driven approach, then retrieves a 3D model candidate that is finally deformed to match the silhouette of the input photo. The probabilistic approach of [16] learns a model that describes the relationship between the parts of 3D shapes which then allows to create an immense variety of new blended shapes by mixing attributes from different models. The sketch driven method of [17] edits a pre-segmented 3D shape using user-drawn sketches of new parts. The sketch is used to retrieve a matching 3D part from a catalogue of 3D shapes which is then snapped onto the original 3D shape to create a new blended 3D shape. Note that the above approaches use only 3D shapes as input for shape blending, with the exception of [15] who use a single photograph and [17] who use sketches. However, unlike ours, neither of these approaches can combine different input images to retrieve a shape that blends parts from each input.

C. MODELING 3D OBJECT PARTS

We will differentiate between 3D segmentation approaches that seek to ensure consistency in the resulting segmentation across different examples of the same object class (co-segmentation) and those that seek a semantically meaningful segmentation (semantic segmentation).

1) CO-SEGMENTATION

Reference [18] learn to segment 3D shapes by fitting a minimal number of primitive shapes (cuboids). While there is no guarantee of semantics in the parts there is a certain consistency in the number and size of cuboids across objects of the same class. Reference [19] detects similarity across object parts by clustering the sub-meshes that constitute each 3D model according to consistency of labels through object parts. Given a large collection of 3D shapes [20] build a network of clusters of shapes of almost equivalent 3D structure, such that the edges connecting neighboring clusters capture a change in structure/topology between them. As all shapes are linked together, the path between two shapes shows the gradual deformations from one shape into another. The projective analysis method of [21] achieves segmentation of 3D objects into semantic parts by back-projecting 2D image labels from a pre-labeled image database onto the 3D model.

2) SEMANTIC SEGMENTATION

The active learning framework of [22] provides accurate semantic region annotations for large geometric datasets with a fraction of the effort. This human-in-the-loop framework alternates between obtaining manual annotations from an expert, automatically propagating the labels to the rest of the dataset and evaluating which are the best examples to label next to minimize both the labeling errors and the human effort. Reference [23] use labeled data from public 3D shape repositories to establish a category specific part hierarchy and label dictionary to then train a hierarchical segmentation and labeling algorithm for 3D shapes. In contrast to these approaches, our semantic part segmentation algorithm works on image data and does not require human interaction at training time.

D. RECOGNITION OF 3D STRUCTURE FROM IMAGES

While our approach can be categorized as shape retrieval, it is closely related to recent work on joint recognition and 3D reconstruction from images. The exemplar-based approach of [24] performs joint object category detection viewpoint estimation, exploiting 3D model datasets to render instances from different viewpoints and then learn the combination of viewpoint-instance using exemplar SVMs. Reference [25] uses 3D Convolutional LSTMs to extract the 3D shape of an object from one or more viewpoints. By using LSTM blocks that contain memory, they progressively refine the shape of the object. Reference [26] learn to generate a 3D point cloud from a single RGB image, it learns purely from synthetic data. By using a point cloud instead of a volumetric representation better definition of the details of the shape are obtained. Their novel approach learns how to generate several plausible 3D reconstructions from a single RGB image at test time if the partial observation of the image is ambiguous.

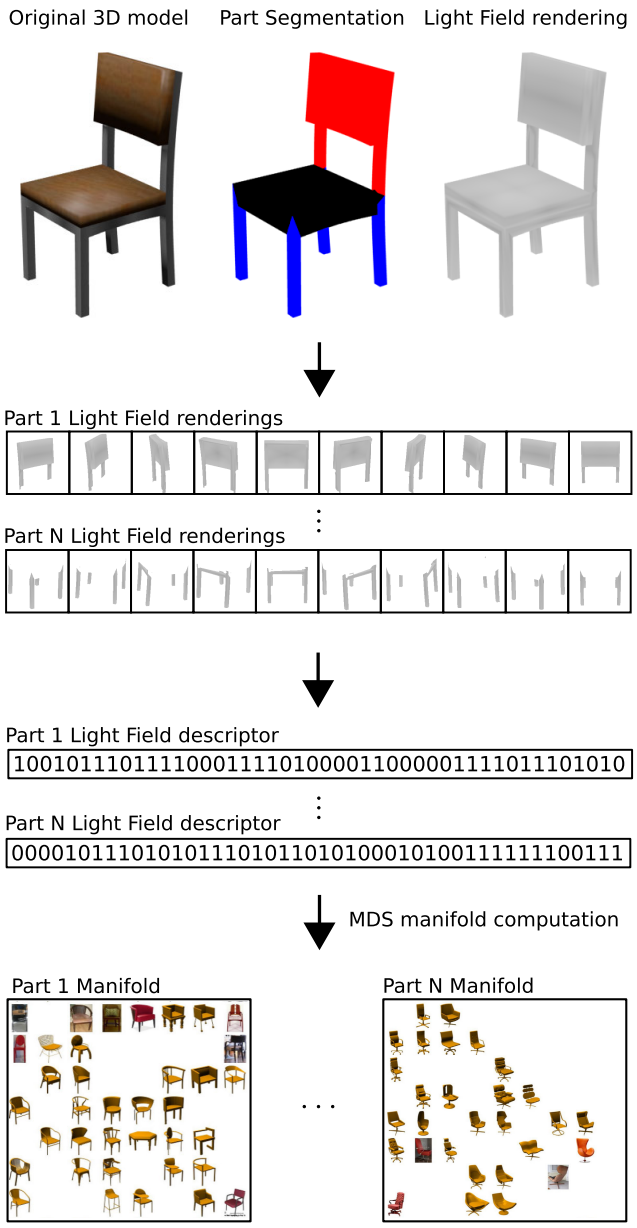

FIGURE 4. Pipeline of the shape embedding construction. The shape of each part of each 3D model is rendered from different viewpoints and represented with a Light Field descriptor [4] which is then characterized with a pyramid of HoG features. The embeddings are then built using non-linear multi-dimensional-scaling (MDS) and the L_2 norm between feature vectors as the distance metric – in each resulting low-dimensional embedding, objects that have similarly shaped parts appear close to each other.

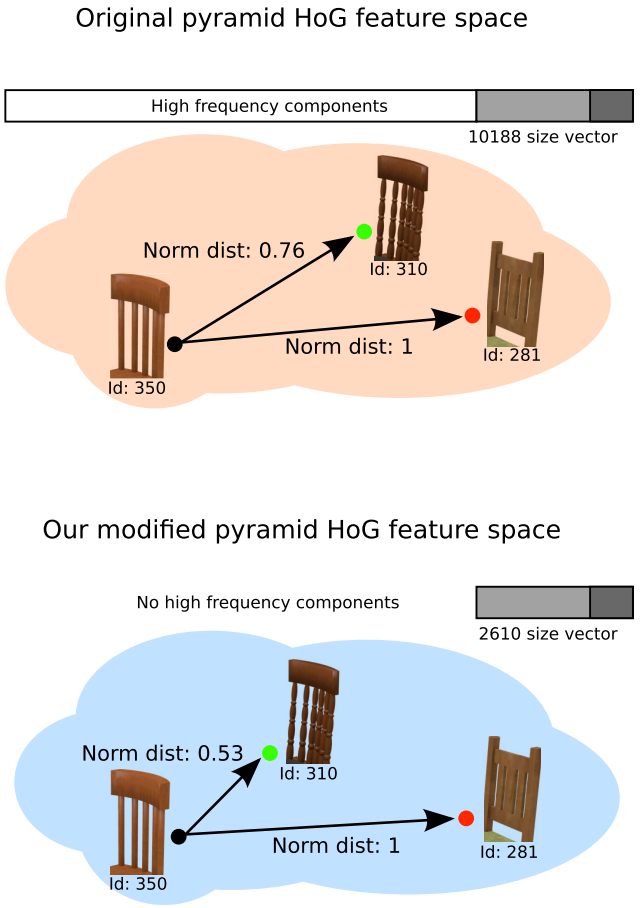

FIGURE 5. Comparison between different HoG pyramid descriptors: (Top) The 3 level pyramid descriptors as used in [2], favors exact matches between shapes since 75% of the components are high frequency responses and dominate the descriptor. (Bottom) Our proposed approach only keeps the lower 2 levels of the HoG pyramid to establish a compromise between detail and smoothness in the similarity metric. We show that the shape with *Id* : 310 is brought closer to the original shape when only using a 2 level pyramid. The distances are more representative of the similarity of 3D structure between the backrests of chairs. The distances have been normalized to offer a correct comparison between both feature spaces.

Reference [27] learn to recognize the object category and the camera viewpoint for an image using synthetically generated images for training. This work showed that datasets of real images annotated with 3D information were not required to learn shape properties from images as this could be learnt from synthetically generated renderings. We take inspiration from this work to train our CNN from synthetic renderings only. Reference [28] obtain good depth estimates for an image given a set of 3D models of the same class. While the goal of our approach is retrieval and not 3D reconstruction, we have taken inspiration from the approaches above in various ways: using 3D model collections, building invariance to viewpoint changes and training from synthetic renderings.

III. OVERVIEW

In this section we provide a high level overview of our **3D Pick'n'Mix** retrieval system. Our system requires a training stage in which: *(i)* embeddings of 3D shapes of object parts are built (see Fig. 4) and *(ii)* a CNN is trained to take as input an image and regress the coordinates of each of its constituent parts on the shape embeddings (illustrated in Fig. 2). At query time the system receives an image or set of images as input and obtains the corresponding coordinates on the part embeddings. If the user chooses object parts from different images a cross-embedding optimization is carried out to retrieve a single shape that blends together properties from different images.

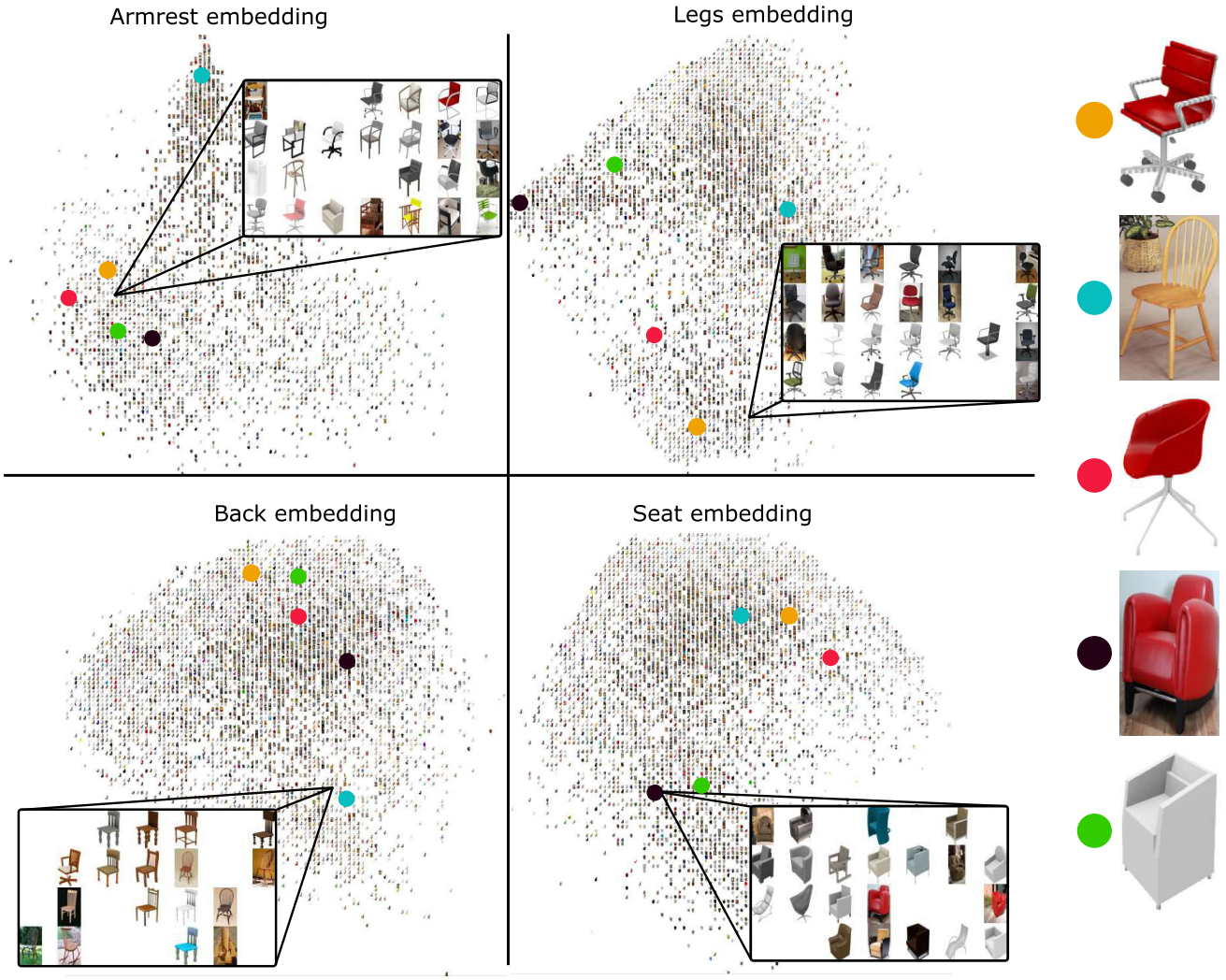

FIGURE 6. Two dimensional visualizations of the four low dimensional part embeddings *"backrest", "seat", "armrests"* and *"legs"* for the chairs class. Probabilistic PCA has been used to provide a 2D visualization of the 128 dimensional embeddings. Both images and 3D models have been represented in the embeddings. Objects with similarly shaped parts lie close to each other on the embedding. Several shapes are tagged in all four embeddings to show how vicinity changes for each part. The figure also highlights that all shapes and images exist in all embeddings, which is what then enables the cross-embedding optimization to retrieve a new shape from the the datasets. The retrieved shape will contain the desired part arrangement.

A. TRAINING

At training time, our method takes as input a class-specific collection of 3D shapes (we used *ShapeNet* [1]) for which part label annotations are available. There are two main steps.

1) EMBEDDING BUILDING

First, our approach learns a separate shape embedding for each object part (see Fig. 4). The shape of each object part is represented with a Light Field descriptor [4] and characterized with a pyramid of HoG features. The embeddings are then built using non-linear multi-dimensional-scaling (MDS) and the L_2 norm between feature vectors as the distance metric – in each resulting low-dimensional embedding, objects that have similarly shaped parts are close to each other. So far these embeddings of object parts (for instance *backrests*, *arm-rests*, *legs*, *seats* in the case of chairs) contain 3D shapes.

2) COORDINATE REGRESSION

The second step at training time is to train a CNN to embed images onto each part embedding by regressing their coordinates. We create a set of synthetic training images with per pixel semantic segmentation annotations for the object parts and ground truth embedding coordinates. The architecture of this novel CNN (which we denote **EmbedNet** and is shown in Fig. 2) has three clear parts: a set of fully convolutional layers for semantic segmentation of the object into parts; a set of convolutional feature extraction layers; and a set of fully connected layers for embedding coordinate regression. This architecture can be trained end-to-end. We give an example of the produced semantic segmentation in Fig. 3.

B. RETRIEVAL

At test time, given a new query image of an unseen object, **EmbedNet** can embed it into each of the part embeddings

by regressing the coordinates. More importantly, our retrieval system can take more than one image as input, picking different object parts from each image. Note that **EmbedNet** only needs the input images and the name of the object part that will be used from each image. The network learns jointly to segment the image into parts and to regress the embedding coordinates and therefore it does not require any manual annotations as input. A **cross-embedding optimization** will then take the coordinates on each of the part embeddings as input and return the coordinates of a unique 3D shape that blends the different object parts together. This is achieved through an energy optimization approach, described in section IV-D.

IV. METHODOLOGY

A. LEARNING SHAPE EMBEDDINGS FOR OBJECT PARTS

While the overall goal of our approach is to obtain a joint representation for images and 3D models, following recent work [2], we choose to create an embedding space that captures the similarity between the shape of object parts based exclusively on the 3D shapes. The reason behind this choice is that 3D models capture a more complete, pure and reliable representation of geometry as opposed to images that often display occlusions, or other distracting factors such as texture or shading effects. We then rely on our new CNN architecture to map images onto the same embedding by regressing their coordinates on the corresponding embeddings.

1) DEFINING A SMOOTH SIMILARITY MEASURE BETWEEN 3D SHAPES

Comparing the 3D shapes of objects from Internet shape collections is often hindered by the fact that meshes might be incomplete, noisy or are not watertight. To avoid these issues we use projective descriptors, more specifically Light-field Descriptors (LfD) [4], in which shape similarity is measured by projective similarity between several corresponding views, since they have been shown to be reliable and particularly well suited for this context in which a large number of shapes that are not watertight need to be compared [2].

We render 20 Light-Field images for each part and build a pyramid of 3 levels of HoG [29] features from each image. The features are then concatenated into a single vector according to their viewpoint [4]. Instead of using all three levels of the pyramid we throw away the descriptors associated with the higher frequencies and only keep the two lowest levels. We found empirically that keeping the high frequency descriptors in the representation will favor exact matches and destroy smooth transitions in shape similarity, as illustrated in Fig. 5.

Shape similarity between object parts is now defined as follows. Given a shape S_i, we define its Light-field Descriptor (LfD) L_i as the concatenation of the HoG responses [29] $L_i = [H_1; H_2; \ldots; H_k]$. The value of k is fixed to $k = 20$ throughout this work. The light field descriptor H_k for each view k is defined as $H_k = [H_k^{mid}; H_k^{low}] \in \mathbb{R}^{2610}$.

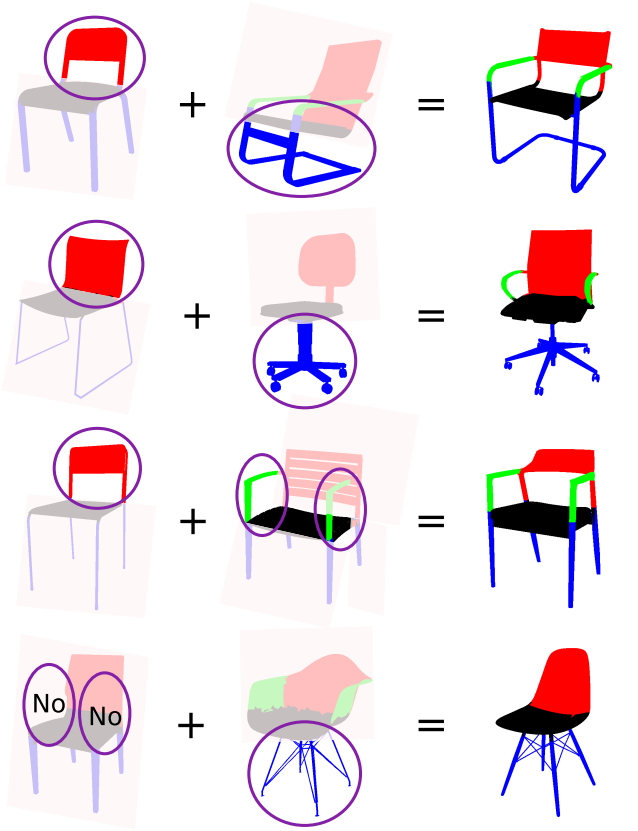

FIGURE 7. Example of shape blending using 3D models as input. The user selects two 3D models as input and chooses the part they would like to keep from each model [note that no annotations are needed, only the name of the part]. The cross-embedding optimization then retrieves the 3D model from the existing shape collection that best fits the arrangement of selected parts. Note that these are actual results from our system and not just examples to illustrate the point.

The L_2 distance between feature vectors is then used as the similarity measure between a pair of shapes S_i and S_j: $d_{ij} = ||L_i - L_j||_2$ where $L_i \in \mathbb{R}^{52200}$. Note that, as described above, we found that using only the mid and low levels of the HoG pyramid leads to smoother transitions in shape similarity. We now build separate embeddings for each object part. Each shape S_i is therefore split into its constituent parts

$$\forall S_i : \exists \{S_i^1; S_i^2; \ldots; S_i^P\} \tag{1}$$

where P is the total number of parts and S_i^p is the shape of part p of object i. If a part is not present in an object (for instance, chairs without arm-rests) we set all the components of the vector L_i^p to zero, which is equivalent to computing the HoG descriptor of an empty image.

B. BUILDING SHAPE EMBEDDINGS OF PARTS

Now that a similarity measure between the shape of object parts has been defined, we use it to construct a low dimensional representation of the shape space. In principle, the original $L_i \in \mathbb{R}^{52200}$ feature vectors could have been used to represent each shape, since distances in that space reflect

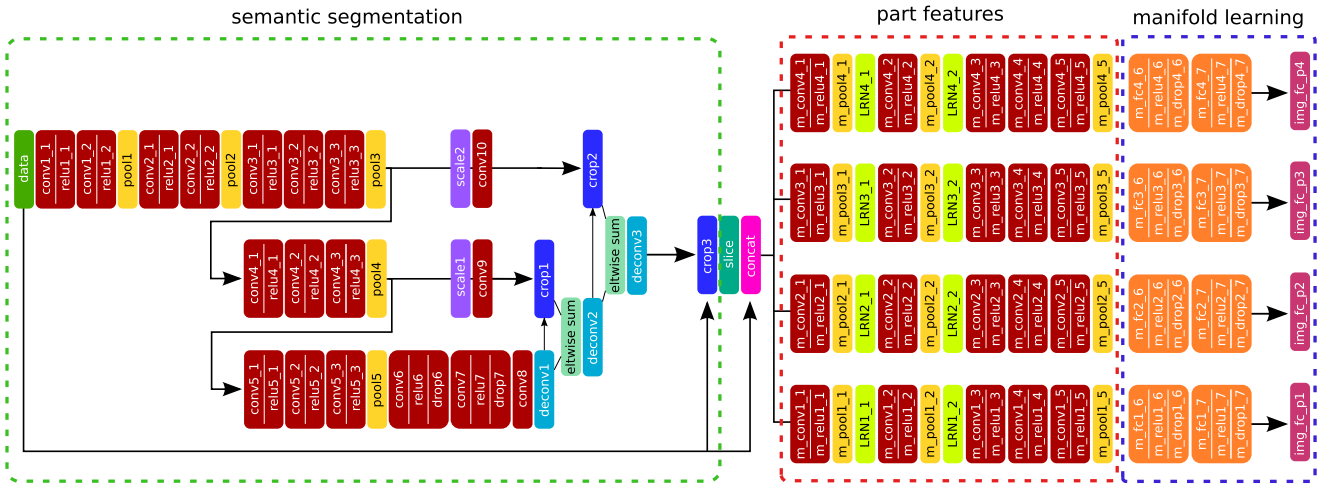

FIGURE 8. Detailed **EmbedNet** architecture with its three well differentiated sections. The first set of layers take care of assigning semantic part labels to each of the pixels in the image. The second stage extracts intermediate features to help the embedding learning. The final stage uses a set of fully convolutional layers to regress the embedding coordinates.

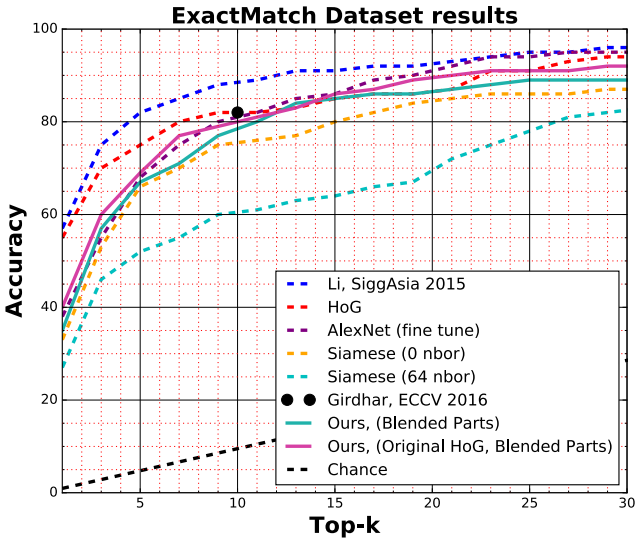

FIGURE 9. Image-based whole shape retrieval results obtained on the ExactMatch dataset. Although our proposed approach is designed to solve a different problem (retrieving object parts, not whole objects) we show that the results are comparable to recent state of the art approaches such as Girdhar ECCV16 [11].

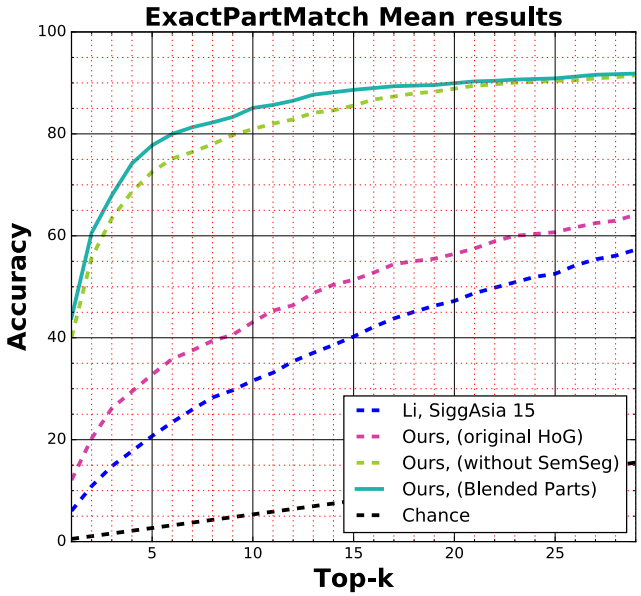

FIGURE 10. Average results of all the parts from Fig.12. We can see that while our approach can still perform quite well when performing whole shape restrieval (Fig.9), methods that model the object as a whole do not perform well when trying to extract a single part. It can also be seen that using the original HoG features works very well when finding the exact the shape, but cannot handle gradual similarity between objects.

well the similarity between shapes. However, as we will use a CNN to map images onto the same joint embedding it is advantageous to reduce the dimensionality of the space, to avoid over-fitting and to speed up the distance computation. We require a much lower dimensional space that still preserves the property that similar shapes lie close to each other on the low dimensional embedding. In practice we reduce the dimensionality from $52,200$ to 128 dimensions and we use non-linear Multi-Dimensional Scaling (MDS) [30] to build the shape embeddings.

We first compute the distance matrix $D^p \in \mathbb{R}^{n\times n}$ as $D^p(i,j) = d^p_{ij}$, were p is the index of the part and n is the total number of shapes. The embedding is built using MDS by minimizing a Sammon Mapping error [31] defined as

$$E^p = \frac{1}{\sum_{i<j} D^p(i,j)} \sum_{i<j} \frac{(D^p(i,j) - D'^p(i,j))^2}{D^p(i,j)}, \qquad (2)$$

where D^p is the distance matrix between shapes in the original high dimensional feature space L^p; and D'^p is the distance matrix between shapes in the new low dimensional embedding L'^p. Different embedding building alternatives could have been selected, like PCA, but as demonstrated in [2], this option best preserves distances between shapes of the original high dimensional feature space.

FIGURE 11. Matrix showing combination results. An experiment was performed that generated all possible combinations of legs and backs from 10 test shapes. This experiment shows that meaningful the results are obtained for all the retrieved blended shapes.

With the different embeddings L'^p for each part p computed, a low dimensional representation of shape similarity exists and all 3D shapes are already included in it. Adding new 3D shapes to the embedding can be done by solving an optimization that minimizes the difference between the distances between all previous shapes and the new shape in D^p and the distances in D'^p with respect to the predicted embedding point. To understand the shape of the produced embeddings we provide a 2D visualization in Fig. 6. In this figure we can better understand how the embeddings relate the different parts of an object.

C. LEARNING TO JOINTLY EMBED IMAGES INTO THE 3D SHAPE LOW DIMENSIONAL SPACE

By building the shape embeddings L'^p for each part p based only on 3D models, we have successfully abstracted away effects such as textures, colours or materials while still encoding viewpoint change due to the Light-field Descriptors. The next step is to train a deep neural network that can map RGB images onto each embedding by regressing the coordinates on each part embedding directly from RGB inputs. Crucially, the input to the network must be simply the RGB image and the name (label) of the object part p selected from that image – for instance *embed this image of a chair into the embedding of "chair legs"*. The CNN must therefore learn first to segment the pixels that depict the chosen object part and then to regress its coordinates on the according shape embedding. The goal is for the mapping onto the embedding to be such that the image appears close to 3D models that share a similarly shaped object part. At this point we want to clarify why we learn a Light Field Descriptor as a proxy of shape. If our task were to learn a metric given RGB inputs we would agree that learning the Light Field descriptor as a proxy would be inefficient. The key of why we learn a Light Field Descriptor is to be able to understand together images and 3D models. Because of the nature of 3D models in big databases the only guarantee is that they will be a polygon soup with textures that you can control. For this scenario the Light Field descriptor has shown in the literature to be a robust way of modelling shape. What we seek to achieve with the neural network is to provide a mapping function from an RGB image to a Light Field Descriptor so we can compare 3D models and images together.

To perform the task at hand, we propose a novel deep learning architecture, which we call **EmbedNet**, that conceptually performs three tasks: first, it learns how to estimate the location of different parts in the image by performing semantic segmentation, it then uses the semantic labeling and the original input image to learn p different intermediate feature spaces for each object part and finally, p different branches of fully connected layers will learn the final image embedding into the respective part shape embedding. It is easy to see that the network has a general core that performs semantic segmentation and specialized branches for each of the embeddings in a similar fashion as the work described in [32].

1) EMBEDNET: A MULTI-EMBEDDING LEARNING ARCHITECTURE

A summary of our new architecture is shown in Fig. 2 and a detailed description of all the layers in Fig. 8. The common part of the architecture, which performs the semantic segmentation, is a fully convolutional approach closely related to [33]. The fully convolutional architecture uses a VGG-16 architecture [34] for its feature layers. A combination of FCN-32, FCN-16 and FCN-8 is used to obtain more detailed segmentations but all sub-parts are trained together for simplicity.

The other two parts of the architecture shown in Fig. 2 take care of: *(1)* creating an intermediate feature space, and *(2)* learning the embedding embedding. The intermediate feature layers take as input the concatenation of the original RGB image and the heat maps given as output by the semantic segmentation layers to learn a feature representation that eases the embedding learning task. Finally, the embedding coordinate regression module is formed by 3 fully connected layers (the first two use relu non-linearities). A dropout scheme is used to avoid over-fitting to the data.

2) DETAILS OF THE TRAINING STRATEGY

The training of such a deep architecture requires careful attention. First, to avoid vanishing gradients and to improve convergence, the semantic segmentation layers are trained independently by using a standard cross-entropy classification loss:

$$L_{seg} = \frac{-1}{N} \sum_{n=1}^{N} \log\left(p_{n,l_n}\right) \tag{3}$$

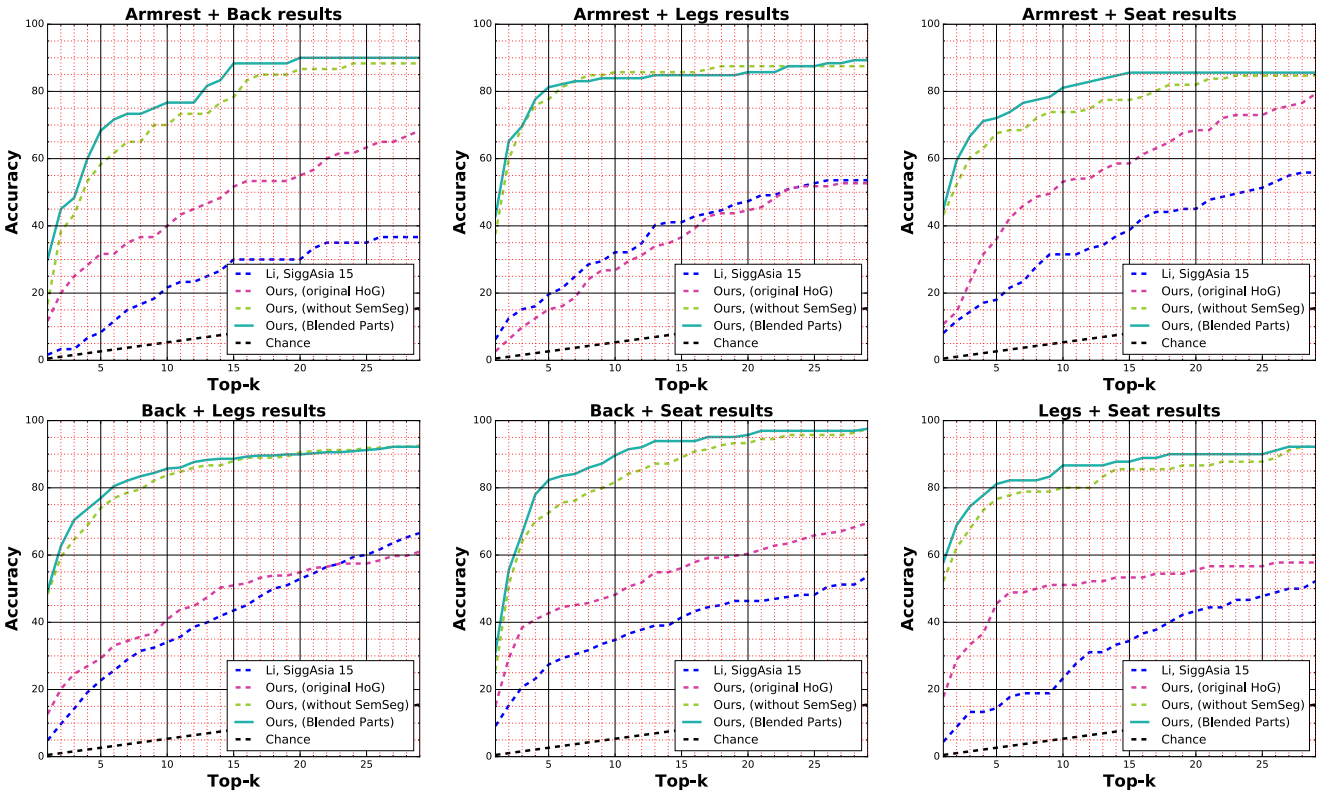

FIGURE 12. Results obtained on the ExactPartMatch dataset when doing shape blending in coordinate space by using embedding estimations for 2 different images to find the correct matching shape. Here we show the individual results for each of the possible combinations of parts.

where p_{n,l_n} is the softmax output class probability for each pixel. A batch size of only 20 is used at this stage due to memory limitations on the GPU and the high number of weights to be trained.

When trying to train the embedding layers we found out that convergence heavily depended on big batch sizes and many iterations. At this point we used a learning scheme that allowed us to have bigger batch sizes during training and faster computation of each iteration. The trick is quite simple really, we precompute for all training images the output of the semantic layers and only train the part branches of the network. By doing this we are training a substantially shallower network allowing for significantly bigger batch sizes. The network is trained by minimizing the following euclidean loss:

$$L_{mani} = \frac{1}{2N}\sum_{i=1}^{N} \|x^i_{est} - x^i_{gt}\|_2^2 \tag{4}$$

where x^i_{est} are the embedding coordinates estimated by the network and x^i_{gt} are the ground truth embedding coordinates. The Euclidean loss is chosen since the part shape embeddings are themselves Euclidean spaces. With this good initialization of the weights we finally perform an end-to-end training of all layers using only the final euclidean loss.

Training Data and Data Augmentation: the training images are generated synthetically by rendering models from *ShapeNet* [1]. We use the 3D part annotations on the 3D models, available from [22], to provide ground truth values for the semantic segmentation. We generate 125 training images per model from different poses, and a random RGB image taken from the Pascal 3D [35] dataset is added as background. An important advantage of the use of synthetic data for training is that generating vast amounts of labeled data from many different viewpoints is cheap, which results in the network being invariant to nuisance factors such as object pose. To recap, the proposed approach is invariant to pose and manages to learn solely from rendered images.

D. SHAPE BLENDING THROUGH CROSS-EMBEDDING OPTIMIZATION

Once the embedding coordinates for the different object parts have been estimated all the information needed to blend them into a single 3D model is available. We formulate this as a retrieval problem: *"find the 3D model, from the existing shapes represented in the embeddings, that best fits the arrangement of parts"*. To be clear, we do not create any new shapes by merging the discovered parts. Instead we solve a retrieval task and search for the 3D shape in the existing collection of models that best satisfies the properties of all the parts. This is largely possible because all the models are represented in all the part embeddings (even when parts are missing).

FIGURE 13. Qualitative results extracted to show performance on different part arrangements using parts from two different image inputs. The results show the closest matching shapes in the Shapenet dataset. If the part is not present, like in the second row, an *X* in the colour of the desired part is used to label the non-presence of that part. It is important to remember that this approach does not require a bounding box of the part, the part is automatically selected and identified by the neural network.

The user selects two (or more) images (or 3D models) and indicates the part they wish to select from each one (note that no annotations are needed, only the name/label of the part). The cross-embedding optimization now finds the 3D shape in the collection that minimizes the sum of the distances to each of the parts.

In more detail – first, all embeddings need to be normalized to allow a meaningful comparison of distances. Then, given the set of embedding coordinates for the selected parts, a shape prediction b can be defined as the concatenation of the respective part coordinates $b = \{b^1; \ldots; b^m\}$. The goal is now to retrieve a 3D model from the shape collection whose coordinates $a = \{a^1; \ldots; a^m\}$ are closest to this part arrangement by minimizing the following distance:

$$B = \min_{a \in \mathbb{S}} \sum_{k=1}^{m} \|a^m - b^m\| \tag{5}$$

where $\mathbb{S}$ is the set of existing shapes. Note that not all parts need to be selected to obtain a blended shape, we define m as

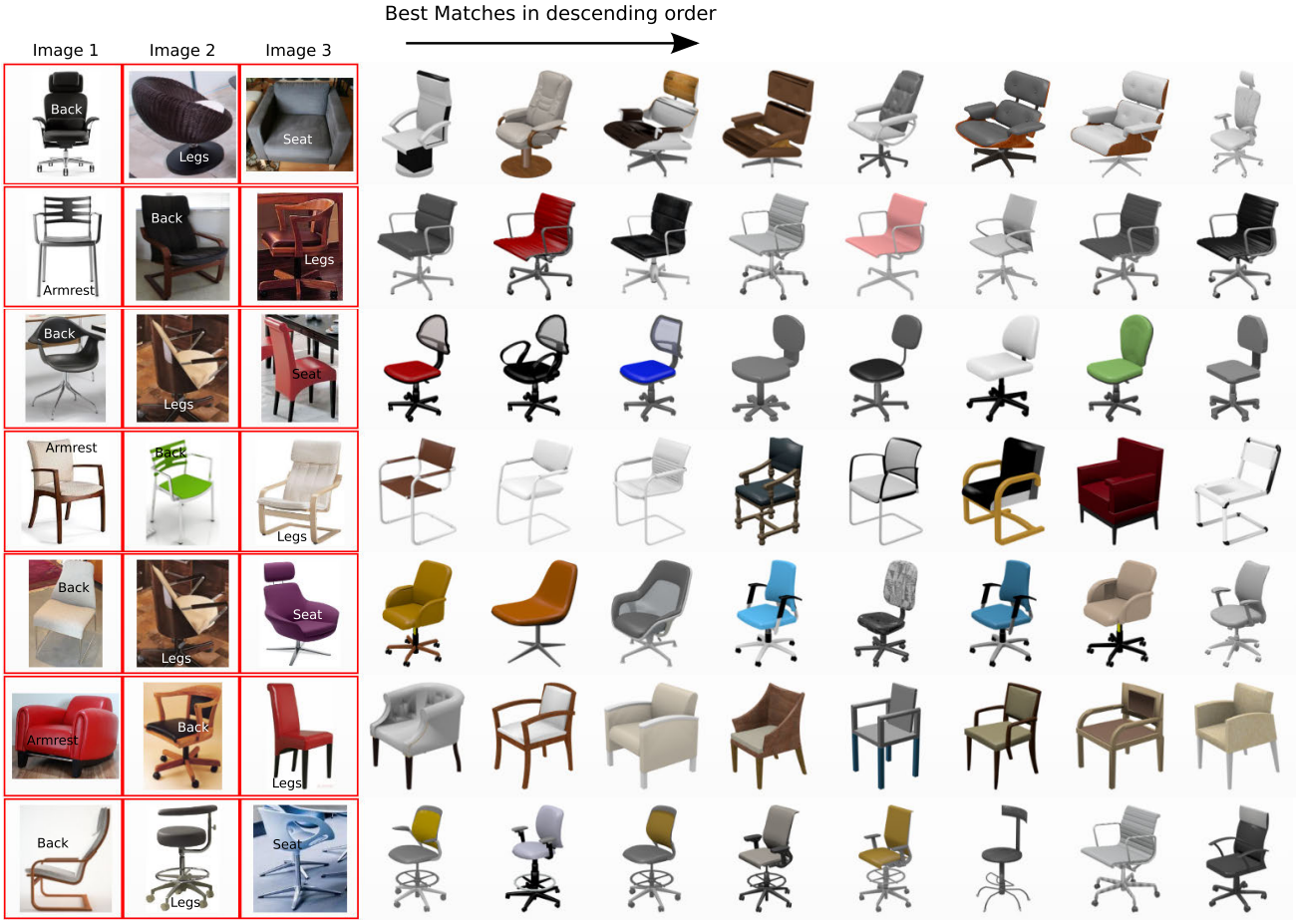

FIGURE 14. Qualitative results extracted to show performance on different part arrangements using parts from three different image inputs. The results show the closest matching shapes in the Shapenet dataset. It is important to remember that this approach does not require a bounding box of the part, the part is automatically selected and identified by the neural network.

the subset of parts to be blended, where $m \subseteq p$. Also, notice that blending can be done by combining any number of parts from any number of sources (shapes/images). A variety of results can be seen in Fig. 11 and Fig. 13. Results with more than two inputs are shown in Fig. 14. Fig. 7 shows an example of shape blending using 3D models as input.

V. RESULTS

We perform a set of qualitative and quantitative experiments to evaluate the performance of our approach. In terms of quantitative evaluation, since there are no equivalent methods that can take more than one image as input to perform shape retrieval and blending of parts, we only carry out comparisons with previous approaches that learn joint embeddings of images and shapes for whole shape retrieval. More precisely we compare our results with [2] on a shape retrieval task since it is the closest work to ours.

We show numerous qualitative results with different part arrangements and a variety of input images and shapes (see Fig. 13, Fig. 14, Fig. 11). We also show results when 3D shapes are used as input (see Fig. 7). For all experiments on real images the neural network has been trained solely using synthetic images.

A. QUANTITATIVE RESULTS ON IMAGE-BASED SHAPE RETRIEVAL

The closest body of research on which this approach can be compared with is the work done in [2]. Both, this approach and [2] can be used as a similarity measure between shapes or images and both have shown to perform well on their specific tasks, the main issue is that while one models parts as separate entities the other models shapes as a whole. This is a big issue because when trying to retrieve the whole shape the proposed approach will be at a serious disadvantage as the inherent probability distributions due to the part arrangements in specific shapes are not lost when modelling the parts individually. On the other hand, when trying to do retrieval of parts [2] will be in a disadvantage as the do not model parts separate. What has been have done is a experimental comparison of both approaches on both tasks, as a clean unbiased comparison cannot be done on a single experiment both approaches will be used to solve both tasks. By doing this how much is lost can be measured in both approaches when either modelling the whole object or the individual parts. To clarify the point, there are no methods that can be directly compared to our approach as we are the first to perform part shape retrieval. The experiment depicted in

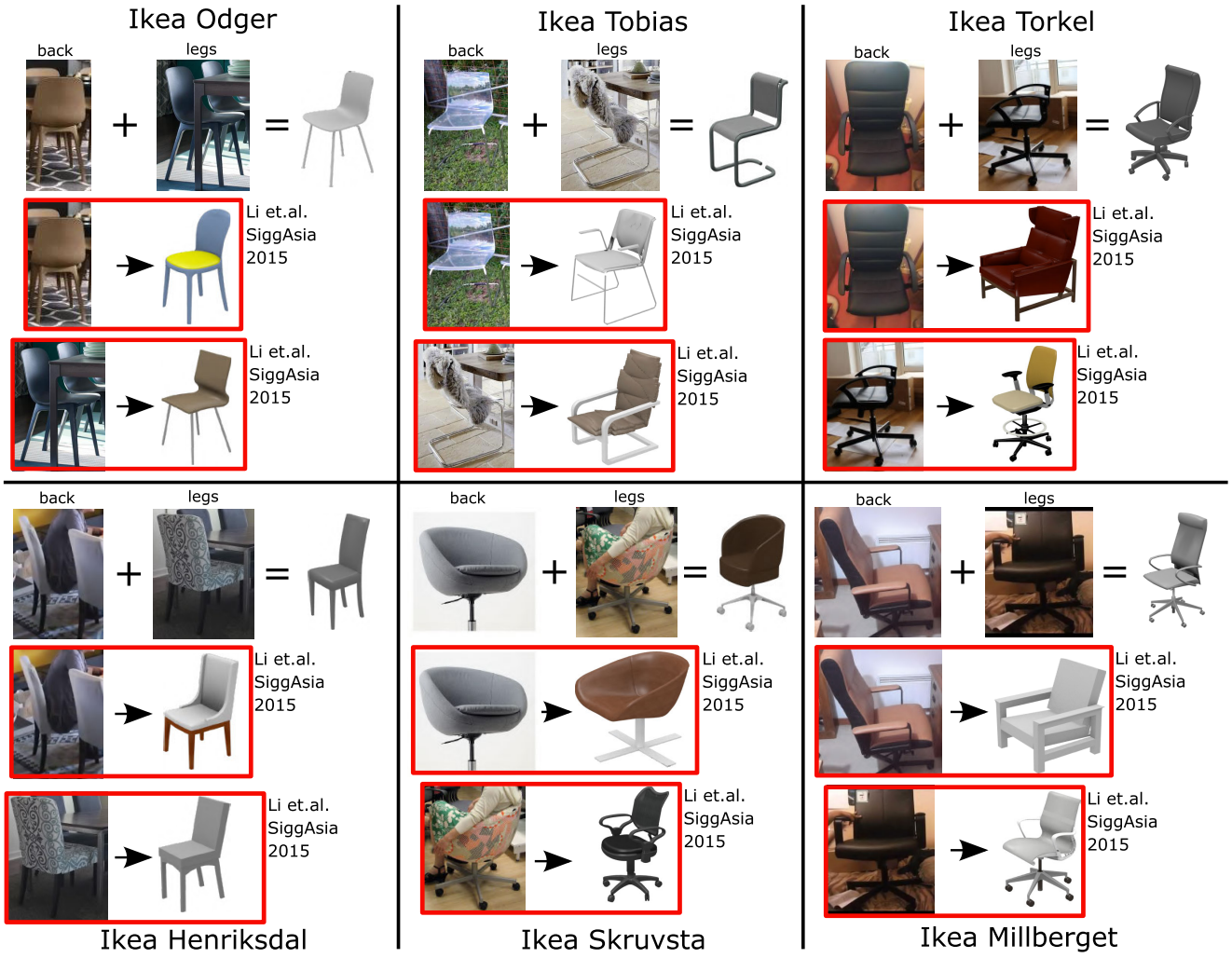

FIGURE 15. Whole shape retrieval from partial observations. Some well known examples of specific chairs that can be found in many households have been selected to show how the proposed approach can obtain the shapes of the parts when occlusion of the rest is present. All images have been obtained from the Internet through a google search. Each example shows the result of performing part retrieval and then shape blending compared to the shape retrieval that can be obtained from [2]. Updated figure.

Fig. 9 and in Fig. 10 are an effort to provide quantitative insights into the performance of our approach with respect the state of the art. By not offering this experiment, with the costly dataset creation involved, and only offering qualitative results one would be unable to asses the real performance of the method. Of course the method presented in this paper is at a disadvantage when doing whole shape retrieval and viceversa when trying to do part shape retrieval. But by cross-comparing the methods performance we show that the presented method is able to perform at a very high level of performance, proving the feasibility of the approach for real applications.

1) IMAGE-BASED WHOLE SHAPE RETRIEVAL

This experiment is the same as performed in [2] in Sec.6.2. The same dataset of images and shapes is used and the experiment is replicated including the previous baselines, the proposed approach and other recent baselines computed on this dataset. The dataset consists of 315 images and 105 shapes, each image has a ground truth shape that corresponds to the chair depicted in the image. The experiment compares against the original results of Li *et al.* [2], Ghirdar *et al.* [11], HoG, AlexNet and Siamese networks. Test samples have not been used during the training of the *EmbedNet* architecture. Two versions of the proposed approach using the original three level HoG pyramid features to build the embedding and the two level HoG embedding features that have been shown to be better fitted for a smoother shape similarity measure. Both approaches based on this work use the same image as input to predict the part coordinates separately in each of the part embeddings. The estimations of each part are used to solve the blending optimization and obtain a single shape prediction from the dataset shapes. The fact that the neural network estimates the coordinates individually means that all the part co-occurrence information that is implicitly encoded in the approaches that model the object as a whole

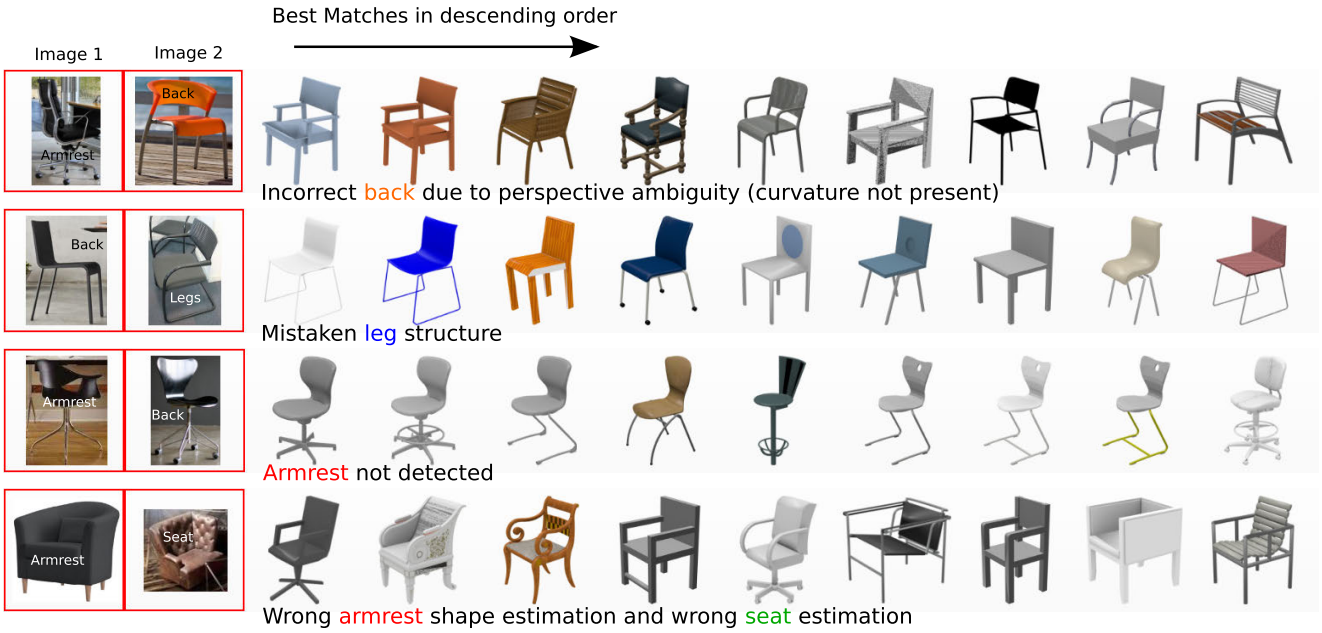

FIGURE 16. Wrong results obtained while trying to blend parts from two different sources. This showcases the limitations of the approach, it can be seen that trying to extract detailed 3D structure in the wild is quite challenging. In many cases the 3*D* structure is unclear due to the perspective of an image, sometimes one of the parts cannot correctly estimate correctly the shape and that generates an incorrect blending and sometimes we are just unlucky and both estimations are incorrect.

is lost during training, nevertheless, the proposed approach can still yield good results that are comparable to those that model the whole shape. It can also be seen that for the exact shape retrieval the original three level pyramid HoG features perform better which is to be expected, albeit, only slightly. The results of these experiments are included in Fig. 9.

2) IMAGE-BASED PART SHAPE BLENDING

As it has been shown, modelling only the parts can still yield good results when retrieving the whole shape. On the other hand, trying to estimate the part similarity using whole shape descriptions does not perform as well. To demonstrate the performance of part retrieval a new dataset has been created using the shapes from the Shapenet database [1] and the images from the ExactMatch Dataset from [2] to create the **Exact*Part*Match** dataset. Cases have been selected in which there is a combination of parts from two different images for which a 3*D* model exists in Shapenet. Test samples have not been used during the training of the *EmbedNet* architecture and the whole dataset is already public online, the link to it will be disclosed after acceptance so the anonymity of the submission is not compromised. The results of these experiments are included in Fig. 10 and the details for each part combination in Fig. 12.

For all approaches similarity is estimated in shape embedding space and then the multi-embedding optimization is performed, which is required to obtain the most similar shape. The approaches shown are *Ours*, with and without semantic segmentation, *Ours* with semantic segmentation but using the original HoG three level HoG features, *Li'sSiggAsia*15 [2] and random chance. It can be seen that [2] struggles to get results as good as the ones obtained by modelling the parts. They have learned a representation of the object as a whole and the embedding neighbourhoods contain much more information than the one needed to detect the correct shape of the part in question. Such holistic representation enables them to better model a whole object but loose substantial performance when trying to identify individually the parts. In contrast the approach outlined in this paper yields much of the information of part co-occurrence appearance in favour of being capable to model the parts individually. It can also see that exploiting the semantic segmentation of the input is consistently better as it defines the actual interest zones in the image. If considering top-5 results without semantic segmentation 65% is obtained but when using semantic segmentation 76% is obtained, which is a substantial 11% improvement in performance. Also using semantic segmentation 90% recall can be obtained at 16 samples while without semantic segmentation 22 samples are required. It can also be seen that trying to blend shapes when using shape similarities that do not correctly model smoothness over shape has a tremendous impact in performance (*OursoriginalHoG*). Chance is shown as a baseline as it is different than the previous experiment due to the number of shapes from which to select the correct one growing to 187.

B. QUALITATIVE RESULTS ON IMAGE-BASED SHAPE RETRIEVAL

To further asses the quality of the results examples are shown depicting the performance of the approach being applied to

real images. In Fig. 13 and in Fig. 14 many example images taken from the **Exact*Part*Match** dataset detailed in previous sections are shown. In this case the entirety of Shapenet needs to be searched and not just the shapes annotated as ground truth. Many different part arrangements are accounted for in the figure to show that the proposed approach can capture not only the big differences but also more subtle differences like the number of legs in the base of a specific swivel chair, the fact that a wooden chair with a back made of bars has a round top or a flat top, capturing the detailed shape of the interconnecting supports of the four legs. In Fig. 16 some of the shortcomings of our approach are outlined. These shortcomings are produced due to that in many cases a projective equivalence of the part exists making its projected shape appearance the same as the one from a different shape, those cases can only be solved by understanding the context of the part arrangement which is something that is lost by modelling the parts individually.

Another experiment is presented in Fig. 11 to help understand how the approach performs on real data. It uses 5 images for the back shape and 5 images for the leg shape, the proposed approach regresses the part shape and then blends the parts in an all against all scheme to show the resulting top matching shapes of each arrangement of parts. It can be seen that the results accurately retrieve a shape that contains the parts from the corresponding images.

C. SHAPE RETRIEVAL FROM PARTIAL OBSERVATIONS

In many cases the object from which we want to obtain its shape is partially occluded and cannot be completely observed. If parts are modelled individually this is no longer an issue as the parts that are visible can be extracted and then several partial observations can be blended together. Examples of this kind of application are shown in Fig. 15

VI. CONCLUSION AND FUTURE LINES OF RESEARCH

The proposed approach has proven to very efficiently manage to capture the part information of objects and is very capable of blending that information to produce a new arrangement of parts that would represent a new shape.

As for the continuation of the research, the proposed approach can be conceived as the encoder part of a generative approach, most of them being based in an encoder-decoder approach, such generative approach could potentially generate new 3D shapes by combining the low dimensional embedding coordinates of each of the inputs. When extending this approach to such a new scenario many problems arise. What kind of oupt would we generate, a voxelized output that models the probability of occupancy of a voxel?. Approaches that generate this kind of output already exist [25], [36], [37], they main drawback that most of these approaches have shown is that they capture the general shape of the object but cannot handle the details of a shape. These approaches generally produce rough descriptions of the desired shape that many times cannot be differenced from another object from the same category.

An alternative to generating a voxel output is to generate a point set as output [26]. Using point sets instead of voxel outputs improves the total volume estimation due to the fact that they use density estimation metrics when predicting the arrangement of the points.

REFERENCES

[1] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*. [Online]. Available: https://arxiv.org/abs/1512.03012

[2] Y. Li, H. Su, C. R. Qi, N. Fish, D. Cohen-Or, and L. J. Guibas, "Joint embeddings of shapes and images via CNN image purification," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–12, Oct. 2015.

[3] M. Hueting, M. Ovsjanikov, and N. J. Mitra, "CrossLink: Joint understanding of image and 3D model collections through shape and camera pose variations," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–13, Oct. 2015.

[4] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung, "On visual similarity based 3D model retrieval," *Comput. Graph. Forum*, vol. 22, no. 3, pp. 223–232, Sep. 2003.

[5] A. P. Sánchez and L. Agapito, "3D pick & mix: Object part blending in joint shape and image manifolds," *CoRR*, vol. abs/1811.01068, Nov. 2018. [Online]. Available: http://arxiv.org/abs/1811.01068

[6] X. Zhang, Z. Zhang, C. Zhang, J. Tenenbaum, B. Freeman, and J. Wu, "Learning to reconstruct shapes from unseen classes," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2018, pp. 2257–2268.

[7] D. Zhang, J. Han, Y. Yang, and D. Huang, "Learning category-specific 3D shape models from weakly labeled 2D images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3587–3595.

[8] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee, "Perspective transformer nets: Learning single-view 3D object reconstruction without 3D supervision," in *Advances in Neural Information Processing Systems*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2016, pp. 1696–1704.

[9] F. P. Tasse and N. Dodgson, "Shape2Vec: semantic-based descriptors for 3D shapes, sketches and images," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 1–12, Nov. 2016.

[10] I. Lim, A. Gehre, and L. Kobbelt, "Identifying style of 3D shapes using deep metric learning," *Comput. Graph. Forum*, vol. 35, no. 5, pp. 207–215, Aug. 2016.

[11] R. Girdhar, D. F. Fouhey, M. Rodriguez, and A. Gupta, "Learning a predictable and generative vector representation for objects," in *Proc. ECCV*, Sep. 2016, pp. 484–499.

[12] T. Funkhouser, M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin, "Modeling by example," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 652–663, Aug. 2004, doi: 10.1145/1015706.1015775.

[13] K. Xu, H. Zhang, D. Cohen-Or, and B. Chen, "Fit and diverse: Set evolution for inspiring 3D shape galleries," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 1–10, Jul. 2012.

[14] I. Alhashim, H. Li, K. Xu, J. Cao, R. Ma, and H. Zhang, "Topology-varying 3D shape creation via structural blending," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 158:1–158:10, Jul. 2014.

[15] K. Xu, H. Zheng, H. Zhang, D. Cohen-Or, L. Liu, and Y. Xiong, "Photo-inspired model-driven 3D object modeling," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 80:1–80:10, Jul. 2011, doi: 10.1145/2010324.1964975.

[16] E. Kalogerakis, S. Chaudhuri, D. Koller, and V. Koltun, "A probabilistic model for component-based shape synthesis," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 55:1–55:11, Jul. 2012, doi: 10.1145/2185520.2185551.

[17] X. Xie, K. Xu, N. J. Mitra, D. Cohen-Or, W. Gong, Q. Su, and B. Chen, "Sketch-to-design: Context-based part assembly," *Comput. Graph. Forum*, vol. 32, no. 8, pp. 233–245, Oct. 2013.

[18] S. Tulsiani, H. Su, L. J. Guibas, A. A. Efros, and J. Malik, "Learning shape abstractions by assembling volumetric primitives," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2635–2643.

[19] O. Sidi, O. van Kaick, Y. Kleiman, H. Zhang, and D. Cohen-Or, "Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering," *ACM Trans. Graph.*, vol. 30, no. 6, pp. 126:1–126:10, Dec. 2011.

[20] N. Fish, O. van Kaick, A. Bermano, and D. Cohen-Or, "Structure-oriented networks of shape collections," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 171:1–171:14, Nov. 2016.
[21] Y. Wang, M. Gong, T. Wang, D. Cohen-Or, H. Zhang, and B. Chen, "Projective analysis for 3D shape segmentation," *ACM Trans. Graph.*, vol. 32, no. 6, pp. 192:1–192:12, Nov. 2013, doi: 10.1145/2508363.2508393.
[22] L. Yi, L. Guibas, V. G. Kim, D. Ceylan, I.-C. Shen, M. Yan, H. Su, C. Lu, Q. Huang, and A. Sheffer, "A scalable active framework for region annotation in 3D shape collections," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 1–12, Nov. 2016.
[23] L. Yi, L. Guibas, A. Hertzmann, V. G. Kim, H. Su, and E. Yumer, "Learning hierarchical shape segmentation and labeling from online repositories," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–12, Jul. 2017.
[24] M. Aubry, D. Maturana, A. A. Efros, B. C. Russell, and J. Sivic, "Seeing 3D chairs: Exemplar part-based 2D-3D alignment using a large dataset of CAD models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3762–3769.
[25] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3D-R2N2: A unified approach for single and multi-view 3D object reconstruction," in *Proc. Comput. Vis. ECCV*, Sep. 2016, pp. 628–644.
[26] H. Fan, H. Su, and L. Guibas, "A point set generation network for 3D object reconstruction from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 605–613.
[27] H. Su, C. R. Qi, Y. Li, and L. J. Guibas, "Render for CNN: Viewpoint estimation in images using CNNs trained with rendered 3D model views," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2686–2694.
[28] H. Su, Q. Huang, N. J. Mitra, Y. Li, and L. Guibas, "Estimating image depth using shape collections," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 1–11, Jul. 2014.
[29] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Washington, DC, USA, vol. 1, Jun. 2005, pp. 886–893, doi: 10.1109/CVPR.2005.177.
[30] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, Mar. 1964, doi: 10.1007/BF02289565.
[31] J. W. Sammon, "A nonlinear mapping for data structure analysis," *IEEE Trans. Comput.*, vol. C-18, no. 5, pp. 401–409, May 1969, doi: 10.1109/T-C.1969.222678.
[32] I. Kokkinos, "UberNet: Training a universal convolutional neural network for Low-, Mid-, and high-level vision using diverse datasets and limited memory," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6129–6138.
[33] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
[34] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015.
[35] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond Pascal: A benchmark for 3D object detection in the wild," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Mar. 2014, pp. 75–82.
[36] A. Kar, S. Tulsiani, J. Carreira, and J. Malik, "Category-specific object reconstruction from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1966–1974.
[37] C. R. Qi, H. Su, M. Niessner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view CNNs for object classification on 3D data," 2016, *arXiv:1604.03265*. [Online]. Available: http://arxiv.org/abs/1604.03265

ADRIAN PENATE-SANCHEZ received the Ph.D. degree from the CSIC-UPC Robotics Lab, Institut de Robòtica i Informàtica Industrial, Barcelona, in 2015, and the Spanish National Scientific Research Council under the supervision of J. A. Cetto and F. Moreno-Noguer. In 2012 and 2013, he was an Idiap Research Institute under the supervision of F. Fleuret. In 2014, he joined the Computer Vision Group at the Toshiba's Cambridge Research Laboratory for a five month internship. From November 2015 till the end of 2017, he was a Research Associate with L. Agapito at UCL working between the fields of 3D computer vision and machine learning. He joined the University of Oxford, in March 2018.

LOURDES AGAPITO received the Ph.D. degree in 1996. She holds the position of Professor of 3D vision in the Department of Computer Science, University College London (UCL). Her research in computer vision has consistently focused on the inference of 3D information from the video acquired from a single moving camera. While her early research focused on static scenes, her attention soon turned to the much more challenging problem of estimating the 3D shape of non-rigid objects (Non-Rigid Structure from Motion, NR-SFM) or complex dynamic scenes where an unknown number of objects might be moving, possibly deforming, independently. Her research group investigates all theoretical and practical aspects of NRSFM deformable tracking, dense optical flow estimation and non-rigid video registration, 3D reconstruction of deformable and articulated structure and dense 3D modeling of non-rigid dynamic scenes. She has held an ERC Starting Grant funded by the European Research Council, from 2008 to 2014. She is also a member of the Vision and Imaging Science Group and the Centre for Inverse Problems.