

# Learning Vector Quantized Shape Code for Amodal Blastomere Instance Segmentation

Won-Dong Jang\*, Donglai Wei, Xingxuan Zhang, Brian Leahy, Helen Yang, James Tompkin, Dalit Ben-Yosef, Daniel Needleman, and Hanspeter Pfister

**Abstract**—Blastomere instance segmentation is important for analyzing embryos' abnormality. To measure the accurate shapes and sizes of blastomeres, their amodal segmentation is necessary. Amodal instance segmentation aims to recover the complete silhouette of an object even when the object is not fully visible. For each detected object, previous methods directly regress the target mask from input features. However, images of an object under different amounts of occlusion should have the same amodal mask output, which makes it harder to train the regression model. To alleviate the problem, we propose to classify input features into intermediate shape codes and recover complete object shapes from them. First, we pre-train the Vector Quantized Variational Autoencoder (VQ-VAE) model to learn these discrete shape codes from ground truth amodal masks. Then, we incorporate the VQ-VAE model into the amodal instance segmentation pipeline with an additional refinement module. We also detect an occlusion map to integrate occlusion information with a backbone feature. As such, our network faithfully detects bounding boxes of amodal objects. On an internal embryo cell image benchmark, the proposed method outperforms previous state-of-the-art methods. To show generalizability, we show segmentation results on the public KINS natural image benchmark. To examine the learned shape codes and model design choices, we perform ablation studies on a synthetic dataset of simple overlaid shapes. Our method would enable accurate measurement of blastomeres in in vitro fertilization (IVF) clinics, which potentially can increase IVF success rate.

**Index Terms**—Blastomere segmentation, Cell segmentation, Amodal segmentation, Shape prior, Vector Quantization, Autoencoder.

## I. INTRODUCTION

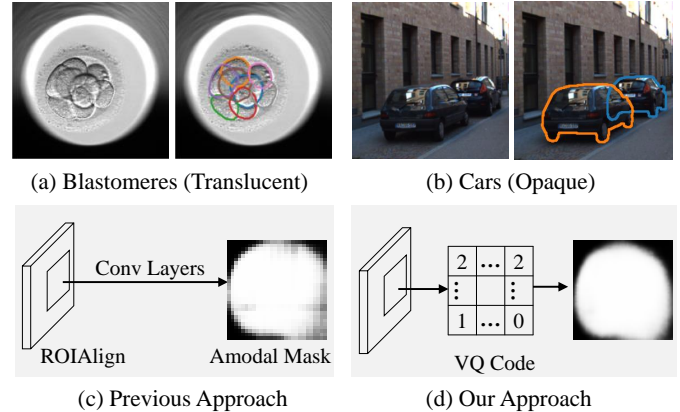
INFERTILE couples worldwide use In-Vitro Fertilization (IVF) to treat their infertility. In a typical IVF treatment, clinicians stimulate the woman to produce many eggs, fertilize those eggs, and culture the resulting embryos for 3–5 days. The clinicians then visually inspect the embryos, select the one that appears most likely to form a viable pregnancy, and transfer it back to the mother. To aid in embryo selection, many modern

W. Jang, D. Wei, B. Leahy, H. Yang, H. Pfister, D. Needleman are with the School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA. corresponding author email: wdjang@g.harvard.edu

X. Zhang is with the School of Engineering, Jiaotong University, China.

J. Tompkin is with the Department of Computer Science, Brown University, Cambridge, MA, USA.

D. Ben-Yosef is with the Tel Aviv Sourasky Medical Center



**Fig. 1. Amodal instance segmentation.** We show an image and its amodal segmentation mask for two common cases: (a) translucent objects overlapping with each other, and (b) opaque objects occluding each other. (c) Previous approaches directly regress the amodal mask from the region of interest (ROIAlign) features. (d) Instead, we first learn a vector quantized (VQ) shape code from ground truth amodal masks, and then classify ROIAlign features into these discrete codes.

clinics employ sophisticated time-lapse imaging systems [1] that record three-dimensional videos of the embryos as they develop.

One feature known to be predictive of an embryo's viability is the shape and symmetry among the cells in the early developing embryo, which are known as blastomeres [2]. However, current clinical practice is to visually score the symmetry at a few distinct points in time, which is time-consuming, inaccurate, and omits much information about the embryo, especially when time-lapse imaging is used. This makes replacing visual symmetry scoring with automated blastomere segmentation a prime candidate for improving clinical IVF practice.

However, while clinics have collected a lot of embryo images from IVF cycles, most existing blastomere segmentation algorithms [3]–[7] use hand-crafted features instead of data-driven approaches. Since hand-crafted methods are tailored to a certain dataset, they may not be robust on different datasets that are collected in varying environments. In this work, we propose a convolutional neural network, which performs amodal visual reconstruction for blastomere segmentation.

Amodal visual reconstruction, predicting the complete shape of partially-visible objects, is part of human ordinary perception. Two common examples of this are: 1) translucent objects visually overlap within the camera's view, such as

when observing biomedical images of cells (Fig. 1a), and 2) opaque objects occlude each other and only a portion of the object is visible, such as when looking down a street at a row of parked cars (Fig. 1b). Beyond its importance in cognitive psychology, amodal visual perception can greatly benefit computer vision applications in practice. With it, biologists can examine new hypothesis through automatic large-scale cell shape measurement from light microscopy images and robotic agents can better navigate through complex environments with partially visible objects.

Unlike the typical instance segmentation setting, which only requires us to label the visible pixels [8]–[11], our wish to predict the shape for invisible or partially-visible object regions requires us to fit a model of shape to the image. Classic solutions have tried known rigid templates of the target object [12], statistical models which capture object shape variation [13], or discriminative parts-based models learned from a dataset [14] potentially with explicit occlusion reasoning [15].

Many recent deep-learning-based models have been proposed for amodal segmentation [16]–[21]. However, these approaches often do not have prior knowledge of the underlying shape, which makes the shape difficult to predict from instance observations under different amounts of occlusion. Further, unlike normal instance segmentation, images of an object under different amounts of occlusion should have the same amodal mask output. Thus, it will be more robust to classify input features into an intermediate robust representation instead of working on the pixel-level.

To exploit this additional information, we propose to learn discrete supervised learning amodal instance segmentation algorithm for partially-visible objects. From binary masks of our object class, we create a deep shape prior as an embedding space with a vector quantized-variational autoencoder (VQ-VAE; [22]). Then, we train our segmentation model to predict the latent representation of an object mask in a bounding box.

Segmentation performance of proposal-based instance segmentation methods [9], [10] highly depends on the bounding box quality. In amodal segmentation, occlusion makes having accurate bounding boxes even more difficult. To tackle this occlusion problem, we add an occlusion detection module to a backbone network. This allows our network to propose better bounding boxes by integrating the occlusion information with the backbone features.

We experiment with a real embryo cell biomedical dataset. Furthermore, we conduct experiments on a synthetic dataset and natural images of street scenes via the KINS dataset [23] to show generalizability of our method. Our approach of encoding objects outperforms state-of-the-art instance segmentation algorithms [9], [21] on both the translucent and occluded types of tested partial visibility.

In summary, our contribution is to propose a novel formulation that incorporates a vector quantized shape code into the amodal instance segmentation pipeline. Additionally, we exploit occlusion information when detecting and segmenting amodal objects via occlusion detection, which can be a new direction for amodal segmentation. This method achieves state-of-the-art performance on not only an internal biomedical

image dataset but also the KINS natural image dataset. Finally, to the best of our knowledge, this is the first approach that applies amodal instance segmentation method to blastomere segmentation.

## II. RELATED WORKS

**Blastomere Segmentation:** Traditional methods predict semantic blastomere masks using hand-crafted features without the instance-level segmentation. Khan *et al.* [7] set seeds inside and outside of cells and optimize Markov random field for segmentation. Rad *et al.* [3] and Kheradmand *et al.* [6] generate blastomere candidates from extracted edges and select the best candidate with in terms of edge coverage. Sidhu and Mills [4] apply thresholding and morphological operations to find the regions of blastomeres and find centers of each cell by measuring distances from pixels to the closest boundary.

Cell-Net proposed by Rad *et al.* [5] is the closest method to ours, training a convolutional neural network for cell localization. However, Cell-Net only predicts blastomere centers, while we perform amodal instance segmentation.

**Amodal Instance Segmentation:** Partially-visible object segmentation is typically studied in biomedical image analysis where cells are often translucent. For nuclei segmentation, Molnar *et al.* [24] fit a circular active contour model [25] using multiple layered distributions of the number of nuclei per pixel. Plissiti and Nikou [26] segment overlapping nuclei by combining nuclei boundary features with priori knowledge of nuclei shape. However, the proposed method works only when given two nuclei centers and requires parameter tuning. Lee and Kim [27] approach translucent cell data as a problem of superpixel segmentation for seed location, and of contour attribution and refinement via graph cuts. Böhm *et al.* [28] segment translucent cell data by learning to lift the image into 3D via a UNet architecture. In both cases, the shape of the object is not specifically represented (e.g., implicitly via a prior), which makes handling occlusion-based partial visibility difficult.

Some works exist on more natural images, e.g., Kihara *et al.* [29] exploit occlusion as a signal to recover full masks for object instances via a Shape Boltzmann machine [30], but not for translucent objects. Li and Malik [16] introduce the first amodal segmentation method. They predict bounding-boxes of modal parts of objects using the object detector [31] and extract segmentation masks using a neural network accepting a pair of an image and a bounding-box as the input. The proposed algorithm iteratively updates segmentation masks by re-computing the bounding-boxes from the output of the network. Zhu *et al.* [17] announce datasets for class-independent amodal segmentation. Multiple subjects annotate the BSDS dataset to analyze the consistency between them. For computational model comparison, they evaluate modal and amodal object proposal algorithms on the proposed amodal COCO dataset. Ehsani *et al.* [18] first perform amodal segmentation and then apply a generative adversarial network to have a complete object image by synthesizing the amodal area. Follman *et al.* [19] predict amodal masks as well as visible masks for occlusion reasoning. Hu *et al.* [20] present a synthetic dataset

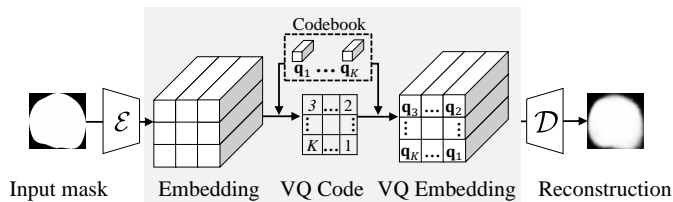


Fig. 2. VQ-VAE architecture containing the mask encoder, embedding quantizer, and mask decoder networks.

for amodal instance level video object segmentation. Qi *et al.* [21] present an amodal segmentation dataset, KINS, by annotating the KITTI detection dataset. They also propose an amodal segmentation network by adding occlusion classification and amodal segmentation branches to the Mask R-CNN framework [9].

Recently, Isack *et al.* [32] introduce the notion of K-convexity, and demonstrated its application in translucent instance segmentation via an energy minimization on an MRF. This allows enforcing a convexity prior on the shape of an instance (such as star [33], geodesic-star [34], hedgehog [35], or regular [36]). However, K-convexity optimization requires seed annotation for each object instance. In contrast, our method learns pixel-wise shape priors and does not require seed annotations.

**Deep Learning for Shape Prior:** These approaches are common in 3D shape completion. Wu *et al.* [37] reconstruct 3D shapes by training shape priors from 3D skeleton parameters; they also later consider the naturalness of reconstructed shapes when training shape priors [38]. Dai *et al.* [39] transform incomplete 3D scans into complete 3D shapes by learning from template shapes. Stutz and Geiger [40] adopt a variational autoencoder [22]. For detection-based instance object segmentation, Kuo *et al.* [41] construct a set of prior masks for each object class and align one of the templates within a bounding box to use it as a shape prior for mask generation. However, deep shape priors are less common for amodal segmentation, where objects overlap each other.

**Deep Learning for Vector Quantization:** Vector quantization methods have been widely used for image compression [42], [43]. Recently, van den Oord *et al.* [44] proposed a vector quantized variational autoencoder for image generation. They show that the proposed method generates more realistic images using learned template codewords. Based on the vector quantized variational autoencoder, Razavi *et al.* [45] developed a hierarchical autoencoder, which encodes an input image in high and low levels. While the high-level codewords contain global information, the low ones have local features. The hierarchical method synthesizes high-quality images by utilizing both global and local information.

### III. VECTOR QUANTIZED SHAPE CODE

Our goal is to learn a discrete representation of amodal shape masks. With it, we can re-formulate the amodal instance segmentation as a classification problem in the low-dimensional latent space. Comparing to previous dense pixel-level mask prediction, the proposed approach can be robust to

occlusion changes and regularized in geometry. To this end, we train a vector quantized variational autoencoder (VQ-VAE) model on the amodal masks to learn the vector quantized (VQ) shape code.

**Comparing Latent Variable Models:** To learn a compact representation of the input, variational autoencoder models (VAE) [22] are commonly used with the Gaussian prior distribution of the latent variable. VAEs learn a global continuous code of the input with the mask encoder model  $\mathcal{E}$ , which can be decoded back for input reconstruction with the mask decoder model  $\mathcal{D}$ . To discretize the learned code, VAE-based clustering methods jointly learn a codebook of embedding vectors that serve as clustering centers. However, as the learned embedding is global, it takes a large codebook for the input to find a similar quantized code. It requires an even larger codebook for a larger number of object categories. VQ-VAEs [44] predict embeddings with spatial resolution and jointly learn a global codebook (Fig. 2). With it, we can use the quantized embeddings to reconstruct input with a limited codebook size.

**VQ-VAE Model:** The key component of VQ-VAE models is the embedding quantizer module. During inference, the mask encoder first transforms the input binary mask  $\mathbf{x}$  into a set of latent vectors  $\mathbf{e}$ . Then, the embedding quantizer assigns each latent vector to the nearest code in the pre-trained codebook  $\{\mathbf{q}_1, \dots, \mathbf{q}_K\}$ . Lastly, the mask decoder transforms the quantized embeddings  $\hat{\mathbf{e}}$  back into a binary mask.

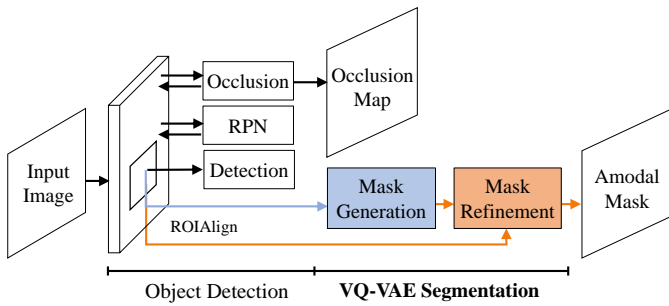
**Learning:** The loss function combines a reconstruction loss, a codebook loss, and a commitment loss. The reconstruction loss is defined as the cross-entropy loss between input mask  $\mathbf{x}$  and the reconstructed mask  $\mathcal{D}(\hat{\mathbf{e}})$ . The codebook loss, which only applies to the codebook, makes the selected codes  $\hat{\mathbf{e}}$  close to the predicted latent vector  $\mathbf{e}$ . The commitment loss, which only applies to the mask encoder, forces the latent vectors  $\mathcal{E}(\mathbf{x})$  to stay close to the matched codes to prevent excessive fluctuations of codes. The full VQ-VAE loss function  $\mathcal{L}_v$  is

$$\mathcal{L}_v = \|\mathbf{x} - \mathcal{D}(\hat{\mathbf{e}})\|_2^2 + \|\mathbf{e} - \hat{\mathbf{e}}\|_2^2 + \beta \|\mathcal{E}(\mathbf{x}) - [\hat{\mathbf{e}}]\|_2^2, \quad (1)$$

where the operator  $[\cdot]$  stands for a stop gradient operation that blocks gradients from flowing into its argument, and  $\beta$  is a hyper-parameter, which is set to 0.25.

**Implementation Details:** The mask encoder has three convolution layers, two residual modules, and one convolution layer. The stride for each convolution layer is 2, which reduces the spatial resolution by half at each layer. For the three convolutional layers, we use 32, 64, and 128  $4 \times 4$  sized filters, respectively. Thus, the mask encoder changes the spatial resolution from  $H \times W$  to  $H/8 \times W/8$ . In the last convolution layer, we set the embedding dimension to 16 empirically. Hence, the mask encoder yields a  $H/8 \times W/8 \times 16$  tensor, which is a set of 16-dimensional latent vectors in embedding space.

For the embedding quantizer, we set the number of codewords  $K$  to 4 empirically, as binary masks are much easier to model than natural images. Also,  $K$  codewords have  $K \times H/8 \times W/8$  possible combinations, which is large enough to model binary object masks.



**Fig. 3. Overview of amodal segmentation pipeline.** We start from an instance segmentation pipeline, e.g., Mask-RCNN. We add the occlusion detection module and replace the original FCN with the proposed VQ-VAE segmentation module. The proposed segmentation model has two steps: initial mask generation through VQ shape code prediction and mask refinement for better localization.

The mask decoder has one convolutional layer, two residual modules, and three deconvolutional layers. Note that each axis of the input image is reconstructed to its original size via the deconvolutional layers. At the end of the decoder, we add a sigmoid layer to constrain values in the reconstructed masks ranging from 0 to 1. The VQ-VAE model is trained separately, and its parameters are fixed after training.

#### IV. AMODAL INSTANCE SEGMENTATION PIPELINE

We propose the VQ-VAE segmentation module to improve amodal instance segmentation. We take the proposal-based instance segmentation approach that contains two modules: object detection and mask prediction (Fig. 3). We attach an occlusion detection branch to object detection (Sec. IV-A) and replace previous fully convolutional network (FCN) with the proposed module for mask prediction (Sec. IV-B). The whole pipeline is trained end-to-end (Sec. IV-C).

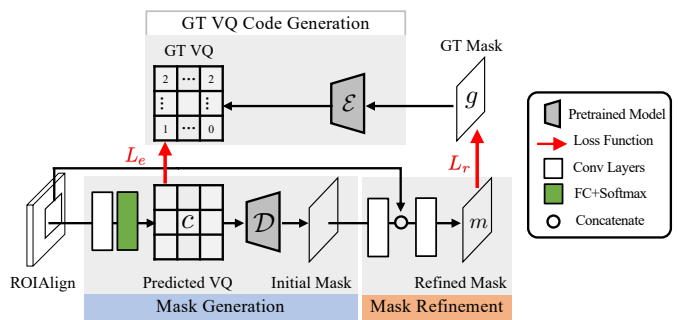
##### A. Object Detection Module

**Backbone:** To extract effective features from the input image, we use ResNet50 [46] as a backbone network, which is trained using the ImageNet dataset [47]. We drop the average pooling layer from the ResNet50 to use spatial features.

**Bounding Box Detection:** The region proposal network (RPN) [31] takes features from the backbone network and measures object existence probabilities and regression parameters of bounding boxes. We employ feature pyramidal networks to extract features across five scales, and minimize the sum of the loss functions at all scales. For each region of interests (ROI), we predict regression parameters and classify its object category.

**Occlusion Detection:** Unlike Faster-RCNN [31], our detection module predicts both bounding boxes and a binary occlusion map. Detecting locations of occlusions allows our object detection module to predict accurate bounding boxes for partially visible objects. Using the backbone features, we estimate probabilities of each pixel being occluded  $\{d_i\}$  via four convolution layers. We adopt the binary cross entropy loss:

$$\mathcal{L}_o = - \sum_{i \in H \times W} \{l_i \log d_i + (1 - l_i) \log (1 - d_i)\}, \quad (2)$$



**Fig. 4. VQ-VAE segmentation module.** We have two segmentation stages: mask generation and mask refinement. We simultaneously minimize the two loss functions,  $\mathcal{L}_e$  and  $\mathcal{L}_r$ .

where  $H \times W$  is the spatial resolution of the backbone feature map and  $l_i$  is the ground-truth occlusion label at pixel  $i$ . We concatenate the output of the second-to-last convolution layer and the backbone feature map to exploit occlusion information in the detection and segmentation modules.

##### B. VQ-VAE Segmentation Module

As shown in Fig 4, the proposed VQ-VAE segmentation module has two steps: initial mask generation and mask refinement. It first generates an initial mask through decoding the predicted VQ-VAE shape code. Then, the refinement step learns to better align the initial mask with the visible object boundaries.

**Initial Mask Generation:** Given the instance-level feature from the object detection module, we first predict the vector quantized shape code and use a pre-trained VQ-VAE decoder model to decode it into object masks with complete shapes.

We first predict a vector quantized shape code instead of a pixel-level binary mask to capture complete shapes using VQ-VAE. We use three convolution layers and one fully connected layer to predict codewords of vector quantized shape code  $c$ . We formulate the problem of vector quantized shape code prediction as a classification problem. For the classification target, we use the pre-trained VQ-VAE mask encoder  $\mathcal{E}$  to encode the ground truth instance mask  $g$  as shown in the right block in Fig 4. One hot encoding makes the encoded mask  $\mathcal{E}(g)$  as a binary representation  $b$ . For the codeword classification at each spatial location, the binary cross entropy loss is defined as

$$\mathcal{L}_e = - \sum_{i \in M \times M \times K} \{b_i \log c_i + (1 - b_i) \log (1 - c_i)\}, \quad (3)$$

where  $M \times M$  is a spatial resolution of a vector quantized shape code and  $K$  is the number of codewords.

We then feed the predicted VQ shape code  $c$  into the VQ-VAE mask decoder  $\mathcal{D}$  to obtain an initial mask.

**Mask Refinement:** The vector quantized shape code can be powerful for shape completion, but the initial mask may not be well-aligned with the detailed object boundary. We add another mask refinement step that combine the instance-level feature and the initial mask feature. To train the refinement decoder,

we set its loss function as

$$\mathcal{L}_r = - \sum_{i \in N \times N \times C} w_i \{g_i \log(m_i) + (1 - g_i) \log(1 - m_i)\}, \quad (4)$$

where  $m_i$  is the probability of a target object occurring at pixel  $i$ .  $N \times N$  is a spatial resolution of the output mask and  $C$  indicates the number of object categories. The weight  $w_i$  is 1 for the channel of the ground-truth object class, otherwise 0.

### C. Learning Strategy

During training, parameters in the region proposal network, detection, mask generation, and refinement modules are updated together to minimize the sum of the loss functions:  $\mathcal{L} = \mathcal{L}_p + \alpha \mathcal{L}_d + \beta \mathcal{L}_o + \gamma \mathcal{L}_e + \delta \mathcal{L}_r$ , where  $\mathcal{L}_p$  and  $\mathcal{L}_d$  indicate the losses for the region proposal network and the detection module, respectively. Hence, we train the proposed network in an end-to-end manner. Empirically, we set the hyper-parameters  $\alpha = \gamma = \delta = 1$  and  $\beta = 0.01$ .

### D. Implementation Details

We provide implementation details of the proposed algorithm including architectures and learning strategies.

**Architecture:** We shrink the spatial resolution of the initial mask using two convolution layers with strides 2. The refinement network consists of four convolution layers and two deconvolution layers. It outputs class-wise masks at each output channel to decouple segmentation and classification.

**Learning:** We initialize parameters in the proposed networks with random values except for the backbone network, which uses weights from the ResNet50 [46]. We train the network via the stochastic gradient descent optimizer. We set the initial learning rate to 0.04, and reduce it to 0.004 and 0.0004 after 10,000 and 11,000 iterations, respectively. We train networks for 12,000 iterations. We use a minibatch size of 16. It takes less than two days to train the proposed networks.

**Running time:** We measure the average computational time of the proposed algorithm on a single Titan X GPU. For images whose shorter axis is fixed to 800, the average running time is 1.75 frames per second. Note that we set the number of proposals to 1,000.

## V. EXPERIMENTS

We compare the proposed method with state-of-the-art methods on a microscopy image dataset and a natural image dataset. Then, we perform ablation studies on the natural dataset to better understand each component and to validate our design choices.

### A. Experiment Setup

**Comparison methods:** For amodal instance segmentation, we can use different object detection pipelines, *e.g.*, Mask-RCNN [9]. With the same pipeline, the proposed VQ-VAE

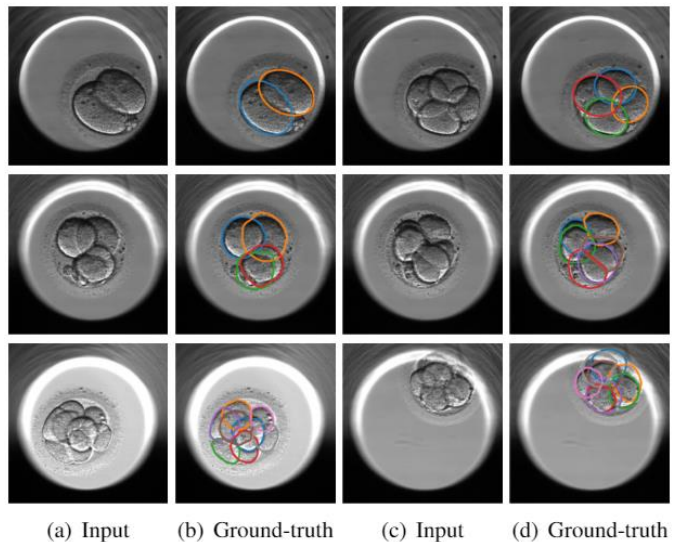


Fig. 5. Examples of cells in the embryo dataset. Cells are delineated by boundaries with different colors.

TABLE I

COMPARISON OF MAP METRIC ON THE EMBRYO CELL DATASET.

Detection	FCN [9]	VQ-VAE (ours)
Mask R-CNN	0.649	<b>0.665</b>

segmentation module is compared with the fully convolutional network (FCN) on two datasets.

**Metrics:** We use mean average precision (mAP), which is standard for object instance segmentation [48]. Let  $AP_k$  denotes a predicted segmentation as correct if its mask intersection over union (IoU) is higher than  $k$ . mAP score is the average of  $\{AP_k\}$  where  $k$  ranges from 0.5 to 0.95 at 0.05 intervals.

### B. Main Results on Embryo Cell Images

In vitro fertilization clinicians predict embryo transfer success by visually observing cell properties like size, granularity, and cleavage (cell split) timing. Cell segmentation of embryo images would automate this property collection for more efficient prediction. Note that our method is more interpretable by clinicians compared to predicting a single number (cell count) from the input image [49].

**Data:** From the IVF clinic in Tel Aviv Medical Center, Israel, we collect 11,671 embryo images, each with a spatial resolution of  $500 \times 500$  pixels. The numbers of cells in each embryo image varies from 2 to 8. Note that we exclude one cell images to evaluate amodal instance segmentation methods. To obtain ground-truth segments, we annotate cells and then ask experts to proofread the annotations. We use 7,054 images for training and the remaining 4,617 for testing. Fig. 5 show examples of embryos and their ground-truth annotations for blastomere instances. We observe that cells are highly overlapping and only partially visible. The size of cells varies as cells cleave and shrink.

**Results:** Table I compares the results of our proposed algorithm with Mask R-CNN [9]. We report mean average

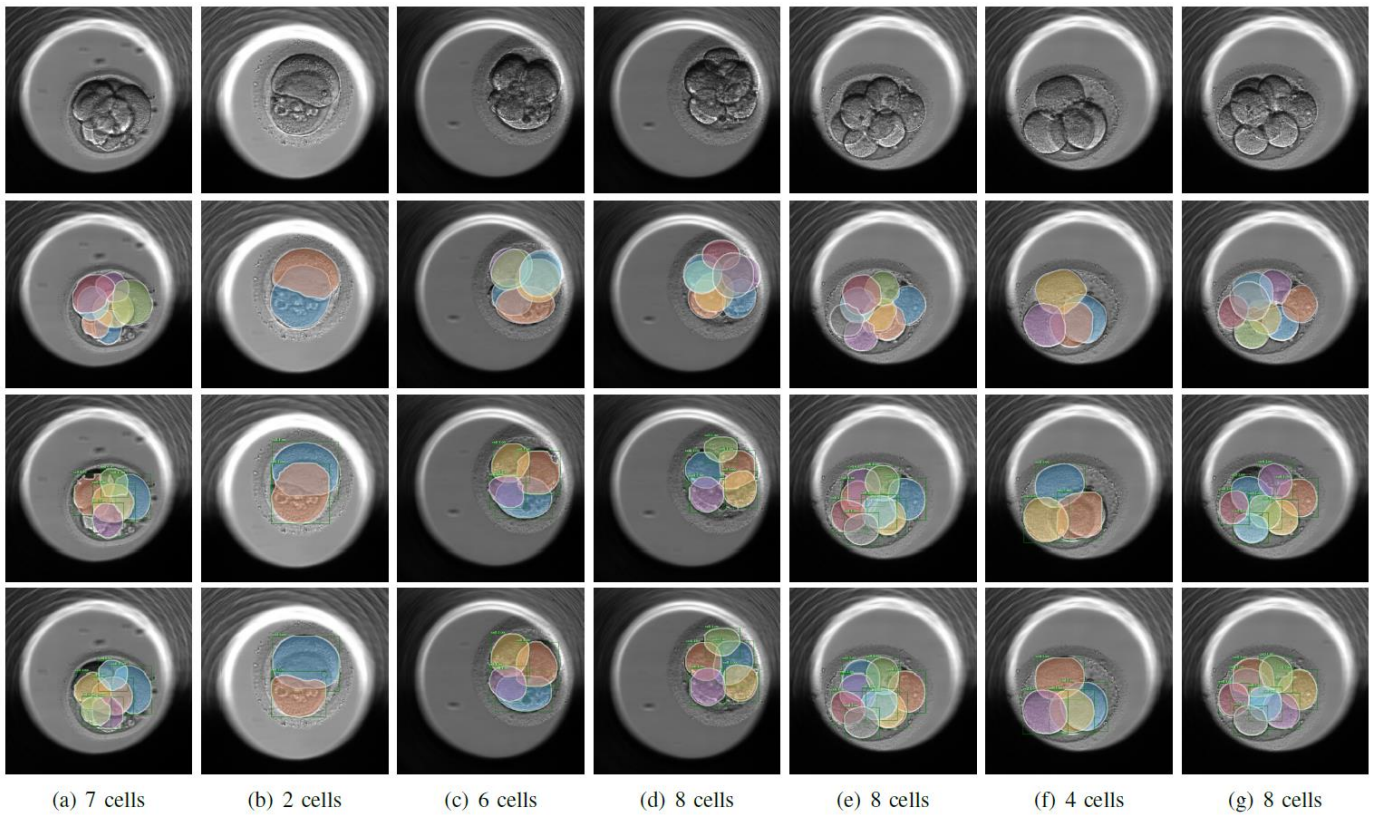


Fig. 6. **Results on embryo cell dataset.** From top to bottom rows, we visualize input images, ground-truth cell masks, and results of FCN and the proposed VQ-VAE, respectively. The segmented object masks are highlighted in coloured regions.

TABLE II

COMPARISON OF mAP INDICES ON THE SYNTHESIZED DATASET.

Detection	FCN [9]	VAE	VQ-VAE (ours)
Mask R-CNN	0.809	0.849	<b>0.865</b>

precision metrics for the evaluation of the cell segmentation methods. The proposed algorithm outperforms the baseline methods. Qualitatively, we observe that the proposed network faithfully detect embryo cells (Fig. 6). Even though partial boundaries of cells are missing, the proposed algorithm generates masks accurately by considering the shape prior of embryo cells.

### C. Additional Results on Synthetic Images

To examine the learned vector quantized shape code and the model design choices, we conduct controlled experiments on a synthetic shape dataset.

**Data:** We synthesize a database of images containing triangles, rectangles, and ellipses. Each image is  $224 \times 224$ , has up to 9 objects in random positions and orientations, with each object set to a random color, and with a random background color. We generate 5,000 training images and 1,000 evaluation images.

**Comparing with VAE:** We use the Mask-RCNN pipeline and compare different shape modeling from FCN (no latent code), VAE (continuous latent code) and VQ-VAE (discrete latent code) modules.

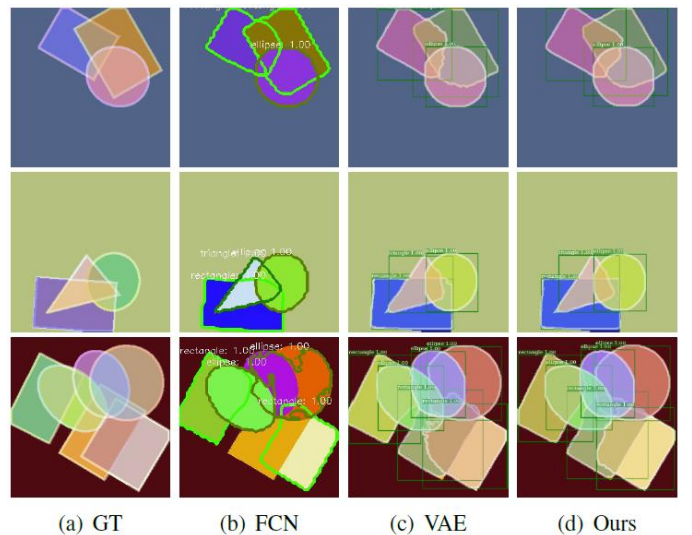


Fig. 7. Qualitative comparison of the proposed algorithm to other mask generation methods on the synthesized dataset. Instances are shown with different colors.

Table II compares mAP indices of the three methods on the synthesized dataset. The proposed VQ-VAE method is better able to delineate complete shapes versus the other two methods. Especially, there is a considerable margin between the proposed algorithm and FCN.

Fig. 7 shows segmentation results for partially-visible object segmentation. The proposed algorithm discovers these objects;

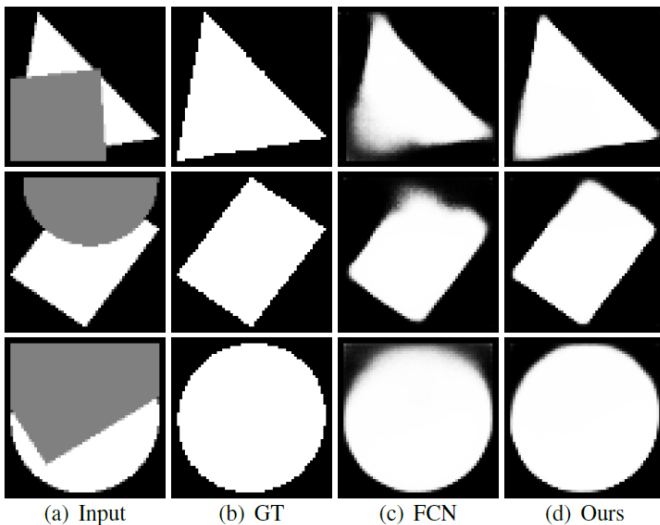


Fig. 8. Comparison of shape completion abilities of our algorithm and FCN.

TABLE III  
COMPARISON OF MAP METRIC ON THE KINS DATASET [21].

Detection	FCN	VQ-VAE (ours)
Mask R-CNN	0.293 [9]	<b>0.303</b>
Mask R-CNN + ASN	0.311 [21]	<b>0.315</b>

though sometimes the baseline methods fail to predict full geometric shapes. We provide more segmentation results in the supplementary materials.

**Robustness to Occlusion:** We assess the occlusion handling abilities of FCN and VQ-VAE when the perfect bounding boxes are given. To this end, we design the following *in silico* psychophysics experiment. Given an input shape, we add another shape in front with different degrees of occlusion. We feed these test images into the trained amodal segmentation models with FCN or VQ-VAE modules.

We quantitatively compare the shape completion performance of the proposed algorithm and FCN. Our algorithm (0.968) outperforms FCN (0.935) in terms of IoU. Moreover, as shown in Fig. 8, the proposed algorithm reliably complete the full shapes of occluded objects. However, FCN makes soft predictions on unseen regions, thus restored regions are blurry.

#### D. Additional Results on Natural Images

To demonstrate the general applicability of our proposed method, we test on an amodal instance segmentation dataset for natural images with a greater diversity of shapes.

**Data:** The KINS dataset [21] is a benchmark for amodal instance segmentation algorithms, which is originally from the KITTI dataset [50]. It consists of 7,474 training and 7,517 test images of driving scenes. The annotated objects belong to one of 7 object classes: pedestrian, cyclist, car, van, tram, truck, and misc-vehicle. The KINS dataset provides both amodal and imodal ground-truth annotations.

**Results:** Table III lists mean average precision metrics of the results of the proposed algorithm with Mask R-CNN [9] and

TABLE IV  
ABLATION STUDY ON THE KINS DATASET [21].

Setting	mAP
VQ-VAE	0.281
VQ-VAE + Refinement	0.298
VQ-VAE + Refinement + Occlusion map	<b>0.303</b>

Mask R-CNN + ASN [21]. Our proposed algorithm performs better than the conventional FCN method on the Mask R-CNN pipeline and yields a slightly better result on Mask R-CNN + ASN. Qualitatively, it finds complete masks of occluded cars (Fig. 9). In the last two rows in Fig. 9, the proposed method fails to segment out cars on the left. This is because the non-maximum suppression removes highly overlapped bounding boxes.

**Ablation Study:** We perform two ablation studies on the KINS dataset. We chose KINS over the embryo dataset for more general analysis, since the objects in KINS have more diverse shapes. We use Mask R-CNN in these studies. First, we remove the occlusion detection branch (VQ-VAE + Refinement). To this end, we train the network without the loss function for occlusion detection  $\mathcal{L}_o$ . Second, we exclude the refinement decoder in the segmentation module (VQ-VAE). To train the network without the refinement decoder, we minimized the embedding loss  $\mathcal{L}_e$  only. We compare these two settings with the full architecture (VQ-VAE + Refinement + Occlusion map) on the KINS dataset. Table IV lists the mAP scores for each ablation setting. Our full architecture performs 0.303 mAP, which is better than the other settings. It indicates that all our components are necessary for accurate amodal segmentation. The inferior performance of the setting without refinement comes from the lack of low-level features.

## VI. CONCLUSION

We proposed an image segmentation method for blastomere instances, which outputs complete masks of cells automatically. The proposed algorithm predicts bounding boxes first and then generates masks. We show that it is effective to learn a mapping from the bounding box features to a shape prior embedding space from a VQ-VAE. This allows us to cope with translucent cells. We also show the benefits of occlusion detection for amodal object detection and segmentation. Our method is applicable for any partially visible objects, not only cells but also geometric shapes, cars, or pedestrians. Experimental results on the embryo, synthesized, and KINS demonstrated that our proposed algorithm outperforms state-of-the-art object instance segmentation methods [9], [21].

Our future works include application to other objects in natural scenes and expanding to biomedical problems that suffer occlusions, such as human blood cell segmentation. We also suggest proposal-free amodal segmentation networks with the center prediction to achieve real-time running speed. Lastly, by adopting generative adversarial networks [51], we might be able to learn shape priors better.



Fig. 9. Results on KINS dataset [23]. The segments are depicted by coloured regions. The object masks are generated using the Mask R-CNN pipeline. The last two rows display failure cases of the proposed method.

## REFERENCES

- [1] S. Armstrong, P. Bhide, V. Jordan, A. Pacey, J. Marjoribanks, and C. Farquhar, "Time-lapse systems for embryo incubation and assessment in assisted reproduction," *Cochrane Database of Systematic Reviews*, no. 5, 2019, publisher: John Wiley & Sons, Ltd.
- [2] C. Racowsky, J. E. Stern, W. E. Gibbons, B. Behr, K. O. Pomeroy, and J. D. Biggers, "National collection of embryo morphology data into Society for Assisted Reproductive Technology Clinic Outcomes Reporting System: associations among day 3 cell number, fragmentation and blastomere asymmetry, and live birth rate," *Fertility and Sterility*, vol. 95, no. 6, pp. 1985–1989, 2011, publisher: Elsevier.
- [3] R. M. Rad, P. Saeedi, J. Au, and J. Havelock, "A hybrid approach for multiple blastomeres identification in early human embryo images," *Computers in biology and medicine*, vol. 101, pp. 100–111, 2018, publisher: Elsevier.
- [4] S. S. Sidhu and J. K. Mills, "Automated Blastomere Segmentation for Early-Stage Embryo Using 3D Imaging Techniques," in *2019 IEEE International Conference on Mechatronics and Automation (ICMA)*, Aug. 2019, pp. 1588–1593, iSSN: 2152-7431.
- [5] R. Moradi Rad, P. Saeedi, J. Au, and J. Havelock, "Cell-Net: Embryonic Cell Counting and Centroid Localization via Residual Incremental Atrous Pyramid and Progressive Upsampling Convolution," *IEEE Access*, vol. 7, pp. 81 945–81 955, 2019.
- [6] S. Kheradmand, P. Saeedi, J. Au, and J. Havelock, "Preimplantation Blastomere Boundary Identification in HMC Microscopic Images of Early Stage Human Embryos," in *arXiv:1910.05972 [cs, eess, q-bio]*, Oct. 2019, arXiv: 1910.05972. [Online]. Available: <http://arxiv.org/abs/1910.05972>
- [7] A. Khan, S. Gould, and M. Salzmann, "Segmentation of developing human embryo in time-lapse microscopy," in *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*. Prague, Czech Republic: IEEE, Apr. 2016, pp. 930–934. [Online]. Available: <http://ieeexplore.ieee.org/document/7493417/>
- [8] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *CVPR*, 2017, pp. 2359–2367.
- [9] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017, pp. 2961–2969.
- [10] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *CVPR*, 2018, pp. 8759–8768.
- [11] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *CVPR*, 2016, pp. 3150–3158.
- [12] T. Knoll and R. Jain, "Recognizing partially visible objects using feature indexed hypotheses," *IEEE Journal on Robotics and Automation*, vol. 2, no. 1, pp. 3–13, 1986, publisher: IEEE.
- [13] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models & their training and application," *Comput. Vis. Image Underst.*, vol. 61, no. 1, pp. 38–59, Jan. 1995, number of pages: 22 Publisher: Elsevier Science Inc. tex.acmid: 206547 tex.address: New York, NY, USA tex.issue.date: Jan. 1995. [Online]. Available: <http://dx.doi.org/10.1006/cviu.1995.1004>
- [14] S. N. Parizi, A. Vedaldi, A. Zisserman, and P. Felzenszwalb, "Automatic discovery and optimization of parts for image classification," in *arXiv*, 2014, arXiv: 1412.6598 [cs.CV].
- [15] J. Winn and J. Shotton, "The layout consistent random field for



- recognizing and segmenting partially occluded objects,” in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, vol. 1, 2006, pp. 37–44, tex.organization: IEEE.
- [16] K. Li and J. Malik, “Amodal instance segmentation,” in *ECCV*, 2016, pp. 677–693, tex.organization: Springer.
- [17] Y. Zhu, Y. Tian, D. Metaxas, and P. Dollár, “Semantic amodal segmentation,” in *CVPR*. IEEE, 2017, pp. 1464–1472.
- [18] K. Ehsani, R. Mottaghi, and A. Farhadi, “SeGAN: Segmenting and generating the invisible,” in *CVPR*. IEEE, 2018, pp. 6144–6153.
- [19] P. Follmann, R. K. Nig, P. H. Rtinger, M. Klostermann, and T. B. Ttger, “Learning to see the invisible: End-to-end trainable amodal instance segmentation,” in *IEEE winter conference on applications of computer vision (WACV)*, 2019, pp. 1328–1336, tex.organization: IEEE.
- [20] Y.-T. Hu, H.-S. Chen, K. Hui, J.-B. Huang, and A. G. Schwing, “SAIL-VOS: Semantic amodal instance level video object segmentation—a synthetic dataset and baselines,” in *CVPR*, 2019, pp. 3105–3115.
- [21] L. Qi, L. Jiang, S. Liu, X. Shen, and J. Jia, “Amodal instance segmentation with KINS dataset,” in *CVPR*, 2019, pp. 3014–3023.
- [22] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in *ICLR*, 2013.
- [23] H. A. Alhaja, S. K. Mustikovela, L. Mescheder, A. Geiger, and C. Rother, “Augmented reality meets computer vision: Efficient data generation for urban driving scenes,” *International Journal of Computer Vision*, vol. 126, no. 9, pp. 961–972, 2018, publisher: Springer.
- [24] C. Molnar, I. H. Jermyn, Z. Kato, V. Rahkama, P. Östling, P. Mikkonen, V. Pietiäinen, and P. Horvath, “Accurate morphology preserving segmentation of overlapping cells based on active contours,” *Scientific reports*, vol. 6, p. 32412, 2016, publisher: Nature Publishing Group.
- [25] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models,” *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, Jan. 1988. [Online]. Available: <https://doi.org/10.1007/BF00133570>
- [26] M. E. Plissiti and C. Nikou, “Overlapping cell nuclei segmentation using a spatially adaptive active physical model,” *IEEE Transactions on Image Processing*, vol. 21, no. 11, pp. 4568–4580, 2012, publisher: IEEE.
- [27] H. Lee and J. Kim, “Segmentation of overlapping cervical cells in microscopic images with superpixel partitioning and cell-wise contour refinement,” in *2016 IEEE conference on computer vision and pattern recognition workshops (CVPRW)*, Jun. 2016, pp. 1367–1373, iSSN: 2160-7516.
- [28] A. Böhm, A. Ücker, T. Jäger, O. Ronneberger, and T. Falk, “ISOODL: Instance segmentation of overlapping biological objects using deep learning,” in *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, Apr. 2018, pp. 1225–1229, iSSN: 1945-8452.
- [29] Y. Kihara, M. Soloviev, and T. Chen, “In the shadows, shape priors shine: Using occlusion to improve multi-region segmentation,” in *CVPR*, 2016, pp. 392–401.
- [30] S. A. Eslami, N. Heess, C. K. Williams, and J. Winn, “The shape boltzmann machine: A strong model of object shape,” *International Journal of Computer Vision*, vol. 107, no. 2, pp. 155–176, 2014, publisher: Springer.
- [31] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in *NIPS*, 2015, pp. 91–99.
- [32] H. Isack, L. Gorelick, K. Ng, O. Veksler, and Y. Boykov, “K-convexity shape priors for segmentation,” in *ECCV*, 2018, pp. 36–51.
- [33] O. Veksler, “Star shape prior for graph-cut image segmentation,” in *ECCV*, 2008, pp. 454–467, tex.organization: Springer.
- [34] V. Gulshan, C. Rother, A. Criminisi, A. Blake, and A. Zisserman, “Geodesic star convexity for interactive image segmentation,” in *CVPR*, 2010, pp. 3129–3136, tex.organization: IEEE.
- [35] H. Isack, O. Veksler, M. Sonka, and Y. Boykov, “Hedgehog shape priors for multi-object segmentation,” in *CVPR*, 2016, pp. 2434–2442.
- [36] L. Gorelick, O. Veksler, Y. Boykov, and C. Nieuwenhuis, “Convexity shape prior for binary segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 258–271, 2017, publisher: IEEE.
- [37] J. Wu, T. Xue, J. J. Lim, Y. Tian, J. B. Tenenbaum, A. Torralba, and W. T. Freeman, “Single image 3D interpreter network,” in *ECCV*, 2016, pp. 365–382, tex.organization: Springer.
- [38] J. Wu, C. Zhang, X. Zhang, Z. Zhang, W. T. Freeman, and J. B. Tenenbaum, “Learning shape priors for single-view 3d completion and reconstruction,” in *ECCV*, 2018, pp. 646–662.
- [39] A. Dai, C. Ruizhongtai Qi, and M. Nießner, “Shape completion using 3D-encoder-predictor CNNs and shape synthesis,” in *CVPR*, 2017, pp. 5868–5877.
- [40] D. Stutz and A. Geiger, “Learning 3D shape completion from laser scan data with weak supervision,” in *CVPR*, 2018, pp. 1955–1964.
- [41] W. Kuo, A. Angelova, J. Malik, and T.-Y. Lin, “Shapemask: Learning to segment novel objects by refining shape priors,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9207–9216.
- [42] E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. V. Gool, “Soft-to-hard vector quantization for end-to-end learning compressible representations,” in *Advances in neural information processing systems*, 2017, pp. 1141–1151.
- [43] L. Theis, W. Shi, A. Cunningham, and F. Huszár, “Lossy image compression with compressive autoencoders,” *arXiv preprint arXiv:1703.00395*, 2017.
- [44] A. van den Oord, O. Vinyals, and others, “Neural discrete representation learning,” in *Advances in neural information processing systems*, 2017, pp. 6306–6315.
- [45] A. Razavi, A. v. d. Oord, and O. Vinyals, “Generating diverse high-fidelity images with VQ-VAE-2,” *arXiv preprint arXiv:1906.00446*, 2019.
- [46] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *CVPR*, 2016, pp. 770–778.
- [47] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *CVPR*, 2009, pp. 248–255, tex.organization: Ieee.
- [48] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in *ECCV*, 2014, pp. 740–755, tex.organization: Springer.
- [49] A. Khan, S. Gould, and M. Salzmann, “Deep convolutional neural networks for human embryonic cell counting,” in *European Conference on Computer Vision workshops*. Springer, 2016, pp. 339–348.
- [50] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? the KITTI vision benchmark suite,” in *CVPR*, 2012, pp. 3354–3361, tex.organization: IEEE.
- [51] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *NIPS*, 2014, pp. 2672–2680.