A STUDY OF SHAPE MODELING AGAINST NOISE

Cheng Long, Adrian Barbu

Florida State University
Department of Statistics
Tallahassee, FL 32306

ABSTRACT

Shape modeling is a challenging task with many potential applications in computer vision and medical imaging. There are many shape modeling methods in the literature, each with its advantages and applications. However, many shape modeling methods have difficulties handling shapes that have missing pieces or outliers. In this regard, this paper introduces shape denoising, a fundamental problem in shape modeling that lies at the core of many computer vision and medical imaging applications and has not received enough attention in the literature. The paper introduces six types of noise that can be used to perturb shapes as well as an objective measure for the noise level and for comparing methods on their shape denoising capabilities. Finally, the paper evaluates seven methods capable of accomplishing this task, of which six are based on deep learning, including some generative models.

Index Terms— shape modeling, shape denoising, object segmentation

1. INTRODUCTION

The analysis of shape has attracted attention for decades since shape is the most basic visual feature of an object. Due to external circumstances, sometimes one cannot obtain any other information such as the color or texture of an object besides the shape. Moreover, other features of an object can be more easily affected by environmental factors. For example, an object's appearance depends on the illumination direction and light color. Hence, shape can be considered a more stable feature of an object, and this is one reason why shape analysis plays an essential role in many computer vision tasks. This paper investigates the problem of restoring object shapes when the shape has been distorted by an external factor, a task that can also be called 'shape denoising'.

To denoise shapes, one needs to use an appropriate shape representation that is flexible enough for this purpose. In earlier studies, different methods were designed to extract shape features such as moments [1], shape context [2], curvature [3], and mathematical morphology [4].

There are different ways to represent shape spaces. Kendall [5] considers shapes as k-tuples of points in $\mathbb{R}^d$. Under this notion of shape space, [6] proposed the Active Shape Model, which performs a PCA on the points of the aligned shapes of the training objects. All shapes are represented using the same number of points, and these points correspond to each other between objects.

In this work, the shape will be represented by a binary image, which is a form of a level set representation. Considering a shape as a binary image allows many generative models to be applied to shape modeling, such as Restricted Boltzmann Machines (RBM) [7], Deep Boltzmann Machines (DBM) [8], Centered Convolutional Deep Boltzmann Machines (CCDBM) [9] and Energy Based Models (EBM) [10]. Moreover, deep learning based methods for semantic segmentation can also be used for shape modeling, e.g. U-Net [11], DeepLabv3+ [12], masked autoencoder (MAE) [13], etc.

1.1. Related Work

The focus of this study is to evaluate and compare the performance of different modeling methods for handling shapes represented by binary images, corrupted by different types of noise. There is a paucity of literature focusing on this topic. However, there is some recent literature containing works about shape modeling.

A recent study about 2D shape modeling [9] introduces convolutions into a DBM-based shape model. The goal of their work is to generate realistic shapes that are different from all training shapes. In their study, the resulting images are generated from test images without any noise. The quality of the generated images is judged subjectively. This evaluation process can not directly describe the shape modeling capability of the proposed method. If the resulting shape is similar to the test shape that has been used for initialization, it only proves that the model can generate results similar to the initial shapes, rather than the ability to represent the entire shape space correctly. If the resulting shape is quite different from the initialization, it is difficult to say whether it is a good shape belonging to the shape space being modeled. In our work, the aim is to recover the object shape from different kinds of noisy perturbations, which allows us to use an objective evaluation criterion: the recovered shape must be close to the original (unseen) shape. In [9], all models used for comparison are based on Restricted Boltzmann Machines. Our work expands the scope of comparison by adding more methods that include both classical ones such as the Active Shape Model(ASM) and recent ones such as the Energy-

based Model(EBM) [10].

In the field of recovering shapes from noisy images, the Active Shape Model (ASM) [6] learns a Point Distribution Model (PDM) from training sets of correctly labeled images with point correspondences, then exploits the linear formulation of the PDM in an iterative search procedure to recover the original shape from a noisy test image. In their study, the authors are more focused on how the parameter of each shape feature influences the final shape. In our work, we pay more attention to the quality of recovered shape and use the IoU (Intersection over Union) between the resulting shape and corresponding ground truth shape for evaluation. More importantly, as we will see in our experiments, the method only works in some cases and with a good initialization. Our works construct many types of noise for a more challenging and realistic evaluation.

In addition to external noise or occlusion, the variation in viewing angles will also increase the shape modeling difficulty. [14] proposed an elastic, affine-invariant shape model to segment images of objects subject to perspective skew. Our work focuses on external shape noise, that needs to be eliminated from objects viewed from a certain range of angles, which should be the same for training and testing.

Since recovering shape from noisy shapes can be seen as a degenerate version of semantic segmentation, the literature in semantic segmentation is related to our work. The Fully Convolutional Network (FCN) [15] converts the classification network [16] into a fully convolutional network by replacing all the fully connected layers with convolutions. FCN has two advantages: it can process input images of any size without losing spatial information and thus reduce computation costs. The U-Net [11] adds a decoder that is symmetrical to encoder based on the FCN architecture and concatenates feature maps from the same level of the encoder and decoder before performing convolutions. DeepLab [17] adopts atrous convolution and fully connected conditional random fields (CRF). In the process of improvement, including DeepLabv2 [18], DeepLabv3 [19] and DeepLabv3+ [12], CRF is deprecated, an atrous spatial pyramid pooling (ASPP) module was proposed.
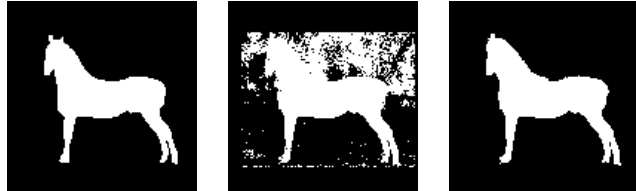
2. THE SHAPE DENOISING PROBLEM

Shape denoising is the process of removing the noise from a shape, with the goal of obtaining a shape as close to the original shape as possible. Figure 1 illustrates an example of shape denoising.

In this chapter we introduce the shape representation that will be used in our study and six different types of noise that can be used to perturb shapes.

2.1. Shape Representation

In this work, shapes will be represented as binary images of a certain size(128×128 in our experiments). All the shape images have black background and white foreground. All foregrounds are centered in the image and have approximately

(a) Original shape   (b) Noisy shape   (c) Denoised shape

Fig. 1. Shape denoising example. The noisy shape (b) has been obtained from the original shape (a) by a noise inducing process such as those described in Section 2.2. A shape denoising method is used to obtain the denoised shape (c).

similar sizes, obtained in an alignment step described below.

Shape alignment. For each binary image I, let $I(x,y) \in \{0,1\}$ be the intensity at location (x,y). Let C_1 be the foreground region, $C_1 = \{(x,y)|I(x,y) = 1\}$ and C_0 the background, $C_0 = \{(x,y)|I(x,y) = 0\}$. The center of mass of the foreground is

$$(\bar{x}, \bar{y}) = \left(\frac{1}{N} \sum_{i=1}^{N} x_i, \frac{1}{N} \sum_{i=1}^{N} y_i \right), \quad (1)$$

where $N = |C_1|$, $(x_i, y_i) \in C_1, i = \overline{1, N}$. Let $d(x_i, y_i)$ be the distance between point $(x_i, y_i) \in C_1$ and the center point $(\bar{x}, \bar{y})$,

$$d(x_i, y_i) = \|(x_i - \bar{x}, y_i - \bar{y})\|_2. \quad (2)$$

Let p_{80} be the 80% percentile of the set $D = \{d(x_i, y_i)|i = \overline{1, N}\}$. The binary image is then rescaled using the scale factor $40/p_{80}$ using bicubic interpolation. The obtained grayscale image is then thresholded to obtain a binary image again, denoted by I'. Let $C_1' = \{(x,y)|I'(x,y) = 1\}$ and $C_0' = \{(x,y)|I'(x,y) = 0\}$ be its foreground and background region respectively. The new center point of the foreground is

$$(\bar{x}', \bar{y}') = \left(\frac{1}{N'} \sum_{i=1}^{N'} x_i, \frac{1}{N'} \sum_{i=1}^{N'} y_i \right) \quad (3)$$

where $N' = |C_1'|$, $(x_i, y_i) \in C_1', i = \overline{1, N'}$. The image I' is then cropped to obtain an image of size 128×128 centered at $(\bar{x}', \bar{y}')$. Padding is used if necessary.

2.2. Introducing Noise in Shapes

We will introduce six types of noise that could be used for perturbing shapes to evaluate the capability of different shape denoising methods in recovering the original shape. The six types of noise are: salt and pepper noise, circle noise, real image noise, occlusion noise, detection image noise and thresholded probability noise, illustrated in Fig. 2.

Salt and pepper noise is obtained by flipping each pixel to its opposite value with a probability p. For this reason it could also be called Bernoulli noise. The flipping probability p is used to control the noise level.

Circle noise is obtained by adding semicircles or punching holes at random locations on the boundary between the foreground and the background. The radius r of the semicircles or holes is used to control the noise level. For each noise image, the radius is fixed.

(a) Original   (b) Aligned   (c) Salt   (d) Circle

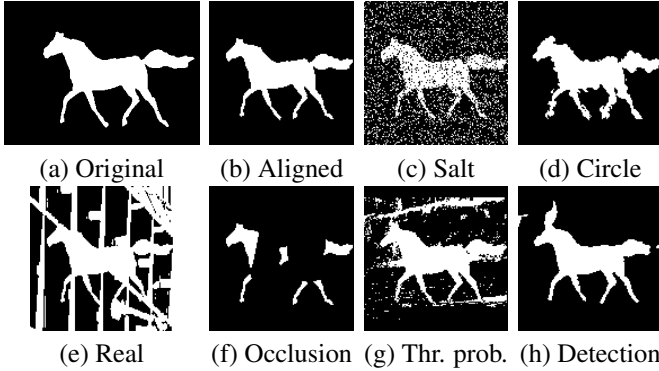(e) Real   (f) Occlusion (g) Thr. prob. (h) Detection

Fig. 2. Illustration of shape alignment and the six types of shape noise introduced in this work. (a) Shape before alignment, (b) Shape after alignment, (c) Salt and pepper noise, (d) Circle noise, (e) Real image noise, (f) Occlusion noise, (g) Thresholded probability noise, (h) Detection image noise.

Real image noise. Real images are binarized by thresholding with various thresholds and the obtained binary image is used to replace the background pixels.

Occlusion noise. Part of the shape is occluded.

Thresholded probability noise. For any binary segmentation M, the corresponding original color image I is used to generate a probability map as follows. Let $I(x, y) \in \{0, 1, ..., 255\}^3$ be the image intensity at location (x, y). Denote N as the number of pixels in the color image, the intensities of all pixels can be represented by a N points in $\mathbb{R}^3$. The N points are clustered into k clusters using k-means clustering, obtaining cluster indices for all pixels $L \in \{1, 2, ..., k\}^N$. Let $C_1 = \{(x, y), M(x, y) = 1\}$ be the foreground region and $C_0 = \{(x, y), M(x, y) = 0\}$ the background region. For each cluster $i \in \{1, 2, ..k\}$, the number of pixels to the foreground region or the background are computed:

$$N_{i1} = |\{(x, y)|L(x, y) = i \wedge (x, y) \in C_1\}| \quad (4)$$

$$N_{i0} = |\{(x, y)|L(x, y) = i \wedge (x, y) \in C_0\}| \quad (5)$$

Then the probability map of color image I can be computed as follows:

$$P(x, y) = \frac{N_{j1}}{N_{j1} + N_{j0}}, \text{ where } j = L(x, y). \quad (6)$$

Then binary noisy shapes are obtained by applying different thresholds to the probability map.

Detection noise. This is the noise introduced during the segmentation of an object from a color or grayscale image. In our experiments, we used a trained Fully Convolutional Network classifier to predict the foreground/background label for the pixels of an image.

2.3. Shape Denoising Evaluation

The capability of a shape modeling method to extract the object shape from a noisy input will be measured using the **Intersection over Union (IoU)**, also known as the Jaccard Index. The IoU is a measure of similarity between two sets, with a range from 0 to 1, with larger values indicating higher similarity. For a binary image I, let C_I be the foreground region, $C_I = \{(x, y)|I(x, y) = 1\}$. For two binary images A, B the IoU is defined as:

$$IoU(A, B) = \frac{|C_A \cap C_B|}{|C_A \cup C_B|} = \frac{|C_A \cap C_B|}{|C_A| + |C_B| - |C_A \cap C_B|} \quad (7)$$

This measure will be used to evaluate the IoU between the denoised shape obtained by a denoising method and the ground truth shape on which the noise was applied to.

3. EVALUATION OF SHAPE DENOISING METHODS

We evaluate seven shape denoising methods (ASM [6], DBM [8], CDBM [9], EBM [10], U-Net [11], Deeplabv3+ [12], MAE [13]) for denoising shapes corrupted by the six types of noise introduced in Section 2.2. The criterion we use to estimate the quality of the denoising result is the IoU (7).

The Weizmann Horse dataset [20] contains 327 horse images and their corresponding mask images. All mask images were aligned as described in section 2.1 to have all shapes centered and of approximately the same size. The aligned images were resized to 128×128. From the 327 aligned images,159 images were randomly selected as the training set S_{train}^{clean} and the other 168 images as the test set S_{test}^{clean}.

The Caltech-UCSD Birds 200 dataset [21] contains photos of 200 bird species. We use 417 images of seven Flycatcher species in our experiment. The images had ground truth segmentations and were aligned as described in section 2.1 and resized to 128×128. From the 417 aligned images, 207 images were randomly selected as the training set S_{train}^{clean} and the other 210 images as the test set S_{test}^{clean}.

Training Sets. For each image in the training set S_{train}^{clean}, noisy versions were generated as follows. **Salt and pepper noise** was generated using 16 flipping probabilities $p \in \{0, 0.01, 0.02, ..., 0.15\}$. **Circle noise** was generated using 11 radii $r \in \{0, 1, 2, ..., 10\}$. **Real image noise** was generated by randomly selecting a 128×128 size random patches from a randomly image of the PASCAL VOC 2012 Dataset. The thresholds to generate binary noise images are $\{10/255, 20/255, 30/255, ..., 250/255\}$.

Test Sets. For each image in the test set S_{test}^{clean}, we use the above methods to generate four types of noisy images. A trained FCN was used to generate detection image noise. We organized all noisy shapes by their IoU with the original shape. For each IOU level (e.g. 0.5-0.6, 0.6-0.7, ...) we randomly selected at most 1000 shapes for testing.

Training details. We used the clean shapes in S_{train}^{clean} to train the models for ASM, DBM, and Convolutional Deep Boltzmann Machines(CDBM). The EBM, U-Net, Deeplabv3+ and MAE were trained with clean shapes as well as shapes perturbed by all types of noise except the detection noise.

Results. The comparison of all methods on the test sets are displayed in Tables 1 and 2. All results that are not significantly worse than the best result in a one-sided paired t-test with $\alpha = 0.05$ are bolded. ASM was not evaluated for the Flycatcher data because there were no manually labeled keypoints available to obtain point correspondences.

Discussion. The experiments reveal that MAE and U-Net are the best shape denoising methods we evaluated for all six

Table 1. Performance (IoU) of the methods evaluated on test sets for the Weizmann Horse dataset.

Input IoU	ASM	DBM	CDBM	EBM	U-Net	Deeplabv3+	MAE
Salt and Pepper Noise							
0.5-0.6	0.476	0.677	0.833	0.881	**0.966**	0.926	0.963
0.6-0.7	0.564	0.704	0.873	0.883	**0.976**	0.934	0.973
0.7-0.8	0.616	0.717	0.893	0.887	**0.983**	0.940	0.982
0.8-0.9	0.629	0.724	0.896	0.893	0.988	0.944	**0.989**
0.9-1	0.653	0.720	0.889	0.895	0.992	0.941	**0.996**
Circle Noise							
0.5-0.6	0.558	0.598	0.588	0.637	0.818	0.751	**0.848**
0.6-0.7	0.600	0.644	0.657	0.705	0.865	0.799	**0.888**
0.7-0.8	0.645	0.685	0.723	0.767	0.897	0.833	**0.914**
0.8-0.9	0.667	0.714	0.797	0.824	0.913	0.865	**0.930**
0.9-1	0.667	0.726	0.882	0.884	0.952	0.915	**0.963**
Real Image Noise							
0.5-0.6	0.492	0.648	0.660	0.805	0.933	0.868	**0.942**
0.6-0.7	0.565	0.683	0.730	0.843	0.954	0.890	**0.963**
0.7-0.8	0.625	0.710	0.790	0.869	0.970	0.910	**0.978**
0.8-0.9	0.644	0.718	0.839	0.886	0.980	0.922	**0.987**
0.9-1	0.657	0.721	0.881	0.895	0.990	0.934	**0.995**
Occlusion Noise							
0.5-0.6	0.285	0.636	0.630	0.706	**0.849**	0.793	**0.852**
0.6-0.7	0.361	0.655	0.670	0.734	**0.875**	0.815	**0.874**
0.7-0.8	0.467	0.665	0.715	0.761	**0.885**	0.829	**0.886**
0.8-0.9	0.559	0.687	0.775	0.801	0.909	0.853	**0.914**
0.9-1	0.643	0.715	0.858	0.870	**0.961**	0.905	**0.964**
Thresholded Probability Noise							
0.5-0.6	0.438	0.623	0.640	0.719	0.839	0.788	**0.845**
0.6-0.7	0.507	0.664	0.712	0.776	0.865	0.813	**0.868**
0.7-0.8	0.577	0.701	0.773	0.823	0.888	0.846	**0.893**
0.8-0.9	0.622	0.714	0.822	0.857	0.915	0.884	**0.917**
0.9-1	0.674	0.732	0.863	0.884	0.939	0.905	**0.941**
Detection Image Noise							
0.5-0.6	0.288	0.632	0.617	0.728	**0.833**	0.749	**0.812**
0.6-0.7	0.486	0.673	0.656	0.743	**0.797**	0.743	**0.803**
0.7-0.8	0.621	0.672	0.728	0.778	**0.805**	0.786	**0.822**
0.8-0.9	0.639	0.701	0.810	0.843	**0.878**	0.856	**0.878**
0.9-1	0.638	0.742	0.869	0.886	**0.930**	0.911	**0.931**

Table 2. Performance (IoU) of the methods evaluated on test sets(Flycatcher in Caltech-UCSD Birds 200 dataset).

Input IoU	DBM	CDBM	EBM	U-Net	Deeplabv3+	MAE
Salt and Pepper Noise						
0.5-0.6	0.627	0.814	0.891	**0.977**	0.942	0.975
0.6-0.7	0.635	0.811	0.893	**0.983**	0.946	0.982
0.7-0.8	0.643	0.816	0.901	**0.988**	0.952	**0.988**
0.8-0.9	0.645	0.812	0.898	0.987	0.953	**0.992**
0.9-1	0.656	0.819	0.909	0.988	0.955	**0.996**
Circle Noise						
0.5-0.6	0.434	0.570	0.542	0.685	0.684	**0.821**
0.6-0.7	0.556	0.662	0.662	0.799	0.778	**0.869**
0.7-0.8	0.626	0.730	0.769	0.871	0.842	**0.906**
0.8-0.9	0.645	0.778	0.830	0.906	0.881	**0.926**
0.9-1	0.649	0.819	0.884	0.938	0.919	**0.942**
Real Image Noise						
0.5-0.6	0.636	0.770	0.852	0.954	0.915	**0.958**
0.6-0.7	0.635	0.785	0.873	0.963	0.925	**0.968**
0.7-0.8	0.638	0.792	0.881	**0.971**	0.932	**0.971**
0.8-0.9	0.650	0.807	0.895	**0.976**	0.939	**0.974**
0.9-1	0.651	0.811	0.901	**0.985**	0.944	**0.983**
Occlusion Noise						
0.5-0.6	0.493	0.408	0.607	**0.863**	0.809	0.845
0.6-0.7	0.540	0.509	0.661	**0.860**	0.815	0.852
0.7-0.8	0.578	0.608	0.717	**0.872**	0.827	0.863
0.8-0.9	0.606	0.705	0.781	**0.886**	0.844	**0.885**
0.9-1	0.630	0.788	0.867	**0.951**	0.908	0.943
Thresholded Probability Noise						
0.5-0.6	0.573	0.610	0.731	0.849	0.796	**0.859**
0.6-0.7	0.587	0.673	0.790	0.870	0.833	**0.878**
0.7-0.8	0.604	0.716	0.828	**0.890**	0.860	0.887
0.8-0.9	0.627	0.759	0.855	**0.910**	0.874	0.897
0.9-1	0.609	0.806	0.836	**0.929**	0.878	0.824
Detection Image Noise						
0.5-0.6	0.518	0.528	0.648	**0.723**	0.703	0.700
0.6-0.7	0.590	0.631	0.751	**0.807**	0.782	0.740
0.7-0.8	0.590	0.697	0.782	**0.815**	0.789	0.772
0.8-0.9	0.653	0.759	0.855	**0.877**	0.865	0.852
0.9-1	0.671	0.800	0.895	**0.925**	0.912	**0.897**

types of noise. DeepLabv3+ is the third best shape denoising method for the six noise types in most situations. EBM outperforms CDBM on all six noise types, especially when dealing with real image noise. In terms of the difficulty of the various types of noise, the salt and pepper noise is the easiest to deal with, followed by real image noise. Circle noise and occlusion noise are more challenging than the above two, especially when the noise level is high. The most challenging noises among these six are the thresholded probability noise and detection image noise.

4. CONCLUSION

This paper introduced the problem of shape denoising, where the shapes are represented as binary images and the goal is to recover a shape that was deteriorated by a noise process. The paper introduced six types of noise that could be used to perturb the shapes. Four of the noise types – real image noise, occlusion noise, thresholded probability noise and detection noise – are related to real challenges encountered during object segmentation.

The goal of this paper is to provide an objective evaluation of shape modeling methods for object segmentation, independent of the image modality they will be used in applications.

The paper evaluated seven methods from different areas that could be used for shape denoising: Active Shape Model (ASM) as a classical segmentation method, two generative models based on Boltzmann Machines (Deep Boltzmann Machine (DBM) and Convolutional DBM), another generative model named Energy Based Model (EBM), and three deep-learning based models used for object segmentation: U-Net, DeepLabv3+ and Masked Autoencoder (MAE).

In the future we plan to apply the trained shape models to object segmentation from color images.

5. REFERENCES

[1] Jan Flusser and Tomas Suk, "Pattern recognition by affine moment invariants," *Pattern Recognition*, vol. 26, no. 1, pp. 167–174, 1993.

[2] Serge Belongie, Jitendra Malik, and Jan Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on PAMI*, vol. 24, no. 4, pp. 509–522, 2002.

[3] Andrei C Jalba, Michael HF Wilkinson, and Jos BTM Roerdink, "Shape representation and recognition through morphological curvature scale spaces," *IEEE Trans. on Image Processing*, vol. 15, no. 2, pp. 331–341, 2006.

[4] Ioannis Pitas and Anastasios N Venetsanopoulos, "Morphological shape representation," *Pattern Recognition*, vol. 25, no. 6, pp. 555–565, 1992.

[5] David G Kendall, "Shape manifolds, procrustean metrics, and complex projective spaces," *Bulletin of the London mathematical society*, vol. 16, no. 2, pp. 81–121, 1984.

[6] Timothy F Cootes, Christopher J Taylor, David H Cooper, and Jim Graham, "Active shape models-their training and application," *Computer vision and image understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[7] Paul Smolensky, "Information processing in dynamical systems: Foundations of harmony theory," Tech. Rep., Colorado Univ at Boulder Dept of Computer Science, 1986.

[8] Ruslan Salakhutdinov and Geoffrey Hinton, "Deep boltzmann machines," in *Artificial intelligence and statistics*. PMLR, 2009, pp. 448–455.

[9] Jiangong Yang, Shigang Liu, and Xili Wang, "Centered convolutional deep boltzmann machine for 2d shape modeling," *Personal and Ubiquitous Computing*, pp. 1–11, 2021.

[10] Bo Pang, Tian Han, Erik Nijkamp, Song-Chun Zhu, and Ying Nian Wu, "Learning latent space energy-based prior model," *arXiv preprint arXiv:2006.08205*, 2020.

[11] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*. Springer, 2015, pp. 234–241.

[12] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018, pp. 801–818.

[13] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick, "Masked autoencoders are scalable vision learners," *arXiv preprint arXiv:2111.06377*, 2021.

[14] Darshan Bryner and Anuj Srivastava, "Bayesian active contours with affine-invariant, elastic shape prior," in *CVPR*, 2014, pp. 312–319.

[15] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015, pp. 3431–3440.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," *NeurIPS*, vol. 25, pp. 1097–1105, 2012.

[17] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:1412.7062*, 2014.

[18] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Trans. on PAMI*, vol. 40, no. 4, pp. 834–848, 2017.

[19] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[20] Eran Borenstein, Eitan Sharon, and Shimon Ullman, "Combining top-down and bottom-up segmentation," in *CVPR Workshop*. IEEE, 2004, pp. 46–46.

[21] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona, "Caltech-UCSD Birds 200," Tech. Rep. CNS-TR-2010-001, California Institute of Technology, 2010.