A Framework to Coordinate Segmentation and Recognition

Wei Huang,[1] Huimin Yu,[1,2]* Weiwei Zheng,[1] Jing Zhang[1]
[1]College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China
[2]The State Key Laboratory of CAD and CG, Zhejiang University, Hangzhou, China,
{huangwayne28, yhm2005, 3090102748, zj9301}@zju.edu.cn

Abstract

A novel coordination framework between the segmentation and the recognition is proposed, to conduct the two tasks collaboratively and iteratively. To accomplish the cooperation, objects are expressed in two aspects: shape and appearance, which are learned and leveraged as constraints to the segmentation so that the object segmentation mask will be consistent with the object regions in the image and the knowledge we have. For the shape, a bottom-top-bottom pathway is built using an encoder-decoder network with capsule neurons, where the encoder extracts the features of the shape that used for recognition and the decoder generates reference shapes according to these features and the recognition result. During this procedure, capsule neurons can parse the existence of the object and cope with the interference in the segmentation. The appearance knowledge is utilized in another pathway to assist the segmentation processing. Both the shape and appearance information are dependent on the recognition result, thus allowing the classifier to convey object information to the segmenter. Experiments demonstrate the effectiveness of our framework and model in collaboratively segmenting and recognizing objects that can be recognized using their shapes/shape-patterns.

Introduction

There has been great progress in fundamental computer vision tasks like fore-/background segmentation and recognition during the past years. In most researches, segmentation and recognition are two separate tasks that have few interactions. The result is that, the segmentation does not necessarily produce a object that can be recognized, and the interpretability of the recognition procedure is not ensured: whether the feature used for recognition truly counts for the present object (e.g., it can be used for reconstructing or generating the object). While for humans, the two tasks are connected to some degree (Vecera and Farah 1997). It's more like a chicken-and-egg problem, where recognition depends on the target segmented out. Segmentation, in turn, count on an understanding of the object. During this procedure, the recognition process has to offer the prior information and knowledge of objects to assist the segmenter.

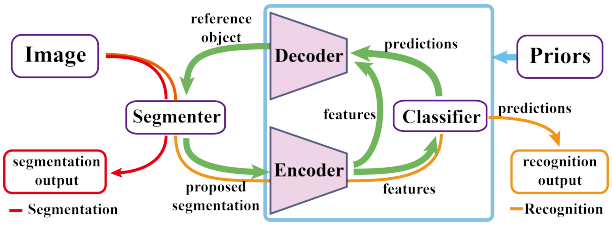

Figure 1: Scheme of coordination between segmentation and recognition. The bottom-up and top-down pathways form an encoder-decoder structure. Other priors may also involve within it.

We believe this perspective of joint segmentation and recognition is worth exploration. Nevertheless, few researchers (George et al. 2017) ever attempted to combine them in object level, despite the effort in semantic segmentation in pixel-level. Therefore, we focus on the cooperation of segmentation and recognition, and put forward with a coordination framework between them, which we hope will bring a novel view to the joint tasks and both tasks can be benefitted from the coordination.

The collaboration between the two tasks requires two pathways: a bottom-up inference procedure to extract features of segmented objects for recognition, and a top-down one which works like a generator, to provide object information (shape, appearance, etc.) from the classifier to the segmenter, according to the prediction and the features. Two pathways comprise an encoder-decoder structure, and the framework of the coordination can be abstracted as Fig.1.

To estabish such a framework, one of the keys is to properly express of the knowledge about objects. Objects can be decomposed into two aspects : (a) shape, describing the global silhouette of the object and (b) appearance (color, texture, etc.), characterizing the local regions inside the silhouette. [1] Many objects have a specific or unique shape or shape pattern that can be recognized. It is much easier to deal with and recognize an object based on its shape than the appear-

*Corresponding author

[1]We use *shape* to represent both the basic shapes, circle, square and etc., and the complex shapes that are composed of these basic shapes. The latter is also referred to as *shape pattern* in this paper.

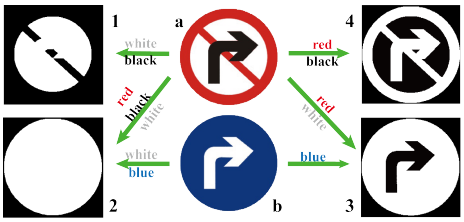

Figure 2: Sign (a) can be recognized using the shape pattern. However, at least four shapes are plausible masks for sign (a), with colored texts indicating the foreground regions. For example, in shape (1), white and black regions are taken as foreground. Conventional models usually segment out the whole like the shape (2). To distinguish (a) from other signs (especially like (b)) and for better discrimination using the shapes, (4) is the best choice while (3) is actually taken as the shape of sign (b). Appearance knowledge is necessary in this case, indicating that white regions in (a) and (b) are the least favorable.

ance, because its appearance can be severely affected by the environments or lighting conditions, which may bring obstacles to the segmentation and recognition. The background is another strong interference if we are dealing with the appearances. Hence, for the top-down generative process, we go for the shape knowledge in the encoder-decoder. Still, in the case where objects have similar shapes or contain multi-regions with different colors, appearance is necessary. For example, in Fig.2, both sign (a) and (b) are consist of several regions and can be recognized using certain shape patterns. The problem is that there might be several possible shape-patterns for a single sign (a). Among them, shape (4) is the most appropriate one for sign (a) so that it is distinguishing from others and can be recognized. However, additional information must be provided to ensure that the segmentation of sign (a) we obtain is (4). In this scenario, shape, appearance, segmentation, and recognition must coordinate with each other. Therefore, we try to fully utilize the appearance knowledge of objects, such as color and texture, based on the training samples. It is leveraged and expressed in another pathway (non-generative, which we call *preference*) in our model, which is dependent on the recognition result.

The overall interaction proceeds as follows: the encoder first extracts the features of the current mask, followed by the recognition. The reference shape is then generated based on both the features and the prediction in the decoder, along with the appearance score map feedback in another pathway. The joint tasks are carried out while the two modules cooperate with each other dynamically, during which noise and interference mainly introduced by the segmenter are inevitable. The encoder-decoder is hence required to be capable of coping with such interference, which is one of the main considerations of our framework. *Capsule* (Sabour, Frosst, and Hinton 2017) neuron is used to address this issue, by analyzing the existence of the truly occurred entity in the present segmentation.

The contributions of our work are:

- Motivated by the interactive processes of human visual mechanism in (Vecera and Farah 1997), we propose a novel framework where object segmentation and recognition can corporate with each other, and make fully knowledge of the objects in both tasks. By considering both tasks at the same time, the segmentation has to be discriminative to humans and computers, and the recognition is much more explanatory.
- In the framework, two aspects of knowledge of object are learned and modeled for joint tasks separately: shape and appearance. Both are dependent on recognition, and used as feedback to guide the segmenter and process the object as a whole in both tasks.
- Capsule neurons are employed in the network to perform shape learning, feature extraction, and recognition, to deal with planar transformations and deformations, as well as the interference during the joint-task evolvement.
- The encoder-decoder architecture in our model can be implemented with numerous different modules. There is a great possibility to extend our model to a more advanced one for collaborative segmentation and recognition.

Relate Work

Variational Segmentation with Shape Priors

Segmentation in the presence of priors has already been widely studied during the past decades. Given an image $\mathbf{I}$, the task can be formulated as the minimization of

$$E(\mathbf{q}) = E_{\text{data}}(\mathbf{I}, \mathbf{q}) + \alpha E_{\text{shape}}(\mathbf{q}) \tag{1}$$

where $\mathbf{q}$ is object mask that can be viewed as the confidence or probability of the pixel inside the object $q(x) \in [0, 1]$. E_{data} denotes the image energy that is designed to describe image energies inside and outside the contour, which also works as a base segmenter. Parametric models like (Mumford and Shah 1989; Chan and Vese 2001) are classic data terms. There are also non-parametric method (Kim et al. 2002), local descriptors (Li et al. 2007; Khan et al. 2015) and etc., to facilitate the model ability in characterizing the objects and background.

The shape term E_{shape} participates as a regularization to constrain the contour based upon the prior shapes $\{\mathbf{q}_i\}$. PCA-based methods were first introduced into the segmentation model in (Leventon, Grimson, and Faugeras 2000) to model the variation of the shapes. Later to avoid making assumptions on the distribution of the prior shapes, (Cremers, Osher, and Soatto 2006) proposed to use kernel density estimation in the shape term to model the shape variation and approximate arbitrary distributions. High-order multiple shape model (Lecumberry, Pardo, and Sapiro 2010), manifold learning (Prisacariu and Reid 2011), sparse representation model (Chen, Yu, and Hu 2013) are also proposed to improve the robustness and flexibility of the shape representations. (Erdil et al. 2016) proposed to use a MCMC sampling method to select proper shapes in $\{\mathbf{q}_i\}$.

Nevertheless, given a series of possible shapes $\{\mathbf{q}_i\}$ of multiple categories, most methods above have limitations on handling the unknown deformations between the observed

shape $\mathbf{q}$ and $\{\mathbf{q}_i\}$. Besides, they obtain the shape term by measuring the similarities between $\mathbf{q}$ and $\mathbf{q}_i$ one-by-one, which humans aren't likely to do. Using a generative procedure to produce reference shape collaboratively may be more in line with human visual cognition mechanism. Several authors (Chen et al. 2013; Kihara, Soloviev, and Chen 2016) proposed to use Restricted Boltzmann Machine to model multiple classes of shapes with global and local deformations and generate reference shapes. The similarity measurement between $\mathbf{q}$ and $\{\mathbf{q}_i\}$ is avoided in these methods.

Simultaneous Segmentation and Classification

Semantic models, such as deep learning models(Long, Shelhamer, and Darrell 2015; Chen et al. 2016), try to perform simultaneous segmentation and classification by dense pixel-labeling. The object is treated as a cluster of pixels in the same category instead of a whole, which usually ignores the overall appearances and shapes of the objects. Some other models try to make structured label prediction by using RBM, like CHOPPS(Li, Tarlow, and Zemel 2013), GLOC(Kae et al. 2013) and MMRBM(Yang, Safar, and Yang 2014). However, the object being recognizable is never under consideration in these models, nor are they capable of recognition. As a result, the segmentation may be too coarse and inaccurate to be discriminative for humans and computers. To segment out and identify each instance, models, like (He et al. 2017), focus on instance segmentation in a multi-task fashion. They do the coordination during the training by optimizing the combined loss, but in the test time, it is hard to tell that there exists coordination between the tasks. One may remove the segmentation branch and the classification maintains the same performance, and/or vice versa, which suggests that the tasks probably won't communicate with each other in the inference time.

The main focus of our work is to coordinate the two tasks, rather than the single segmentation or recognition task. We aim to combine object-level classification and segmentation and make them interact with each other. To fully utilize the knowledge of objects is another concern of our work.

Capsule Neuron

A capsule (Sabour, Frosst, and Hinton 2017) is a multi-dimension vector neuron $\mathbf{v}$, where each element v_i represents a property (e.g, scaling, angle, etc.) of the corresponding entity and the length $||\mathbf{v}|| \in [0, 1]$ represents the probability that entity exsits in the image. It is introduced to capture the geometric and spatial relationships between the object-part entities from low-level to high-level. Only when most low-level object parts can agree on the existence of a high-level part, then the corresponding high-level capsule gets a higher $||\mathbf{v}||$. This property of capsules differs from that of a standard CNN, and enables them to parse the objects that suffer from heavy interference, while a normal CNN basically only analyzes the existence of these features. The spatial correlations are discarded during the forward inference in a normal CNN. The consequence is that given a jumbled object (for instance, a face with misplaced eyes, nose, and mouth), a conventional CNN may recognize it as a normal object. What is more important is that a normal CNN cannot recover the object when there is much interference. Experiments have illustrated the effectiveness of the capsules in modeling multiple objects with various deformations and segmenting overlapping objects.

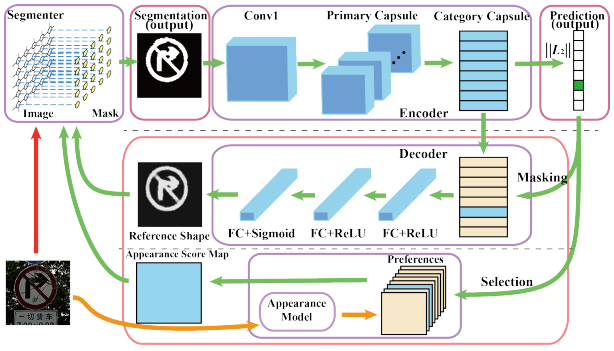

Figure 3: Framework of the proposed model.

Another interesting feature of capsule neurons is that each dimension of the features encoded with the capsule neurons represents a specific property or local part of the object and is interpretable to the computers and even humans (Fig.4 in (Sabour, Frosst, and Hinton 2017)).

Proposed Model

Our model is illustrated in Fig.3. For clarity, we first describe the notations. Given an image $\mathbf{I}$ with height H and width W, $\mathbf{q} \in [0, 1]^{H \times W}$ denotes the object mask, and $c \in \{1, 2, \cdots, L\}$ denotes the object label, both to be solved in the optimization. $\mathbf{V} = \text{Enc}(\mathbf{q})$ is the encoder in the capsule network, which outputs L category capsules grouped as $\mathbf{V} \in \mathbf{R}^{L \times D}$ and each with D elements, while $\text{Dec}(*)$ the decoder or reconstruction subnetwork, which takes label prediction c and category capsule data $\mathbf{V}$ as input. $\tilde{\mathbf{q}}_i = \text{Dec}(i, \mathbf{V})$ denotes the reconstruction using i-th category capsule data $\mathbf{v}_i$ (others masked out during reconstruction, as in (Sabour, Frosst, and Hinton 2017)). $w_i = ||\mathbf{v}_i||$ is the existence probability of a category i entity, and $\{\mathbf{s}_i\}_{i=1,\cdots,L}$ represents the *preferences* (see the *Appearance Constraint* section) for each category.

Compared to Eq.1, the objective function of our model consists of three terms:

$$E(\mathbf{q}, c) = E_{\text{data}}(\mathbf{I}, \mathbf{q}) + (\alpha E_{\text{shape}}(\mathbf{q}, c) + \beta E_{\text{appearance}}(\mathbf{q}, c)) \quad (2)$$

where the first term describes the energy produced by the present image and segmentation, and the other two describe the constraint given by the present segmentation and prior knowledge of objects.

Shape Constraint

Classic shape models are incapable of generating samples. RBM can, but it requires an external classifier to do prediction. Moreover, the procedure to infer all the layers jointly is usually intractable as RBM gets deeper, which makes RBM less scalable, practical and popular. These drawbacks have limited the performance of RBMs in recognition tasks.

Another candidate for E_{shape} is the standard CNN autoencoder network. The problem is that a conventional CNN encoder cannot parse the object when there's a lot of interference. The features of the object and the interference entangle with each other, which is then fully utilized for reconstruction. Therefore, the reference shape produced by a standard CNN network still contains much interference and is much less instructive and meaningful. Since the interference will surely emerge during the joint tasks, a conventional CNN encoder is insufficient for our framework.

For a trained capsule network with dynamic routing, given an object with interference, capsules will determine whether the object exists and what its pose is. Noise, interference and irrelevant parts can be filtered and only features of the objects that is believed to exist will contribute to the reconstructed reference shape. The capsule network is incredibly suitable for the bottom-up parsing processing.

Based on the extracted features $\mathbf{V}$ by the capsule network, the reference shapes can be generated by $\{\tilde{\mathbf{q}}_i = \text{Dec}(i, \mathbf{V})\}_{i=1,\cdots,L}$. A simple shape constraint term then can be defined using the present segmentation $\mathbf{q}$ and the prediction c:

$$E_{\text{shape}}(\mathbf{q}, c) = \sum_{i=1}^{L} I(i = c)\text{CE}(\mathbf{q}, \tilde{\mathbf{q}}_i) \tag{3}$$

where $\text{CE}(\mathbf{x}, \mathbf{y}) = -\sum_j (x_j \ln(y_j) + (1 - x_j) \ln(1 - y_j))$ is the cross entropy, and $I(\cdot)$ the indicator, $c = \arg\max_i w_i = \arg\max_i ||\mathbf{v}_i||$. Notice that $\tilde{\mathbf{q}}_i$ is depending on the category i as well. An alternative for Eq.3 is a weighted version, which may be better than a hard assignment in Eq.3 in the early stage of the segmentation (because the segmentation in the beginning may be a mess):

$$E_{\text{shape}}(\mathbf{q}, c) = \sum_{i=1}^{L} \frac{w_i}{\sum_{k=1}^{L} w_k} \text{CE}(\mathbf{q}, \tilde{\mathbf{q}}_i) \tag{4}$$

Appearance Constraint

If we were to see a white elephant, the first glimpse may surprise us. We probably won't consider it's something non-elephant after that (instead we may think it gets the albinism). This intuition suggests that for human, shapes are more relied on than appearances when they make recognitions on objects, considering that appearances can be affected severely by the environments and circumstances (light, materials, etc.). Nevertheless, many objects, like traffic signs, human skins, and some animals usually contain some certain colors or specific textures. These are auxiliary knowledge about objects that can contribute to the joint tasks.

Currently we utilize the appearance based on the present image and scenario, rather than generating one, because generating real colored objects is much complicated and remains an open question (Kingma and Welling 2013; Goodfellow et al. 2014; Larsen et al. 2015), while it's much simpler to generate a shape (the fore- and background are meanwhile partitioned). *Preference* expresses such knowledge by assigning different biases on different regions or parts of objects, depending on object categories. It can be viewed as something like "attention", popular in visual processing : some parts should be more focused on than others.

For image $\mathbf{I}$, preferences can be obtained by applying L linear classifiers to the pixels, cliques or superpixels. For category i, its linear classifier is trained by regarding the regions of the objects with label i as positive and others negative, including background and regions of objects in other categories. Suppose $\mathbf{U}_i$ is the weight for the i-th linear classifier, then the preference for the pixels in region Ω_{p} according to class i is, $\mathbf{s}_i^{\Omega_{\text{p}}} = \mathbf{U}_i \mathbf{z}^{\Omega_{\text{p}}}$, where $\mathbf{z}^{\Omega_{\text{p}}}$ is the feature of region Ω_{p}. Given an image $\mathbf{I}$, its class-dependent preferences are then $\{\mathbf{s}_i\}_{i=1,\cdots,L}$ and thus the appearance term is as follows:

$$E_{\text{appearance}}(\mathbf{q}, c) = -\sum_{i=1}^{L} I(i = c)\mathbf{q}^T \mathbf{s}_i \tag{5}$$

Same as the shape term, the appearance term as well relies on the present recognition c and the segmentation $\mathbf{q}$. The minimal E_{aprc} (*aprc* is short for *appearance*) is obtained with the right segmentation $\mathbf{q}$, the correct recognition c and the corresponding appearance $\mathbf{s}_c$. With E_{aprc}, we then make the appearance influential to the recognition c^{t+1} by evaluating the energy with present segmentation $\mathbf{q}^t$, the old label c^t and the new prediction y:

$$c^{t+1} = \begin{cases} y & E_{\text{aprc}}(\mathbf{q}^t, y) \leq \rho E_{\text{aprc}}(\mathbf{q}^t, c^t) \\ c^t & E_{\text{aprc}}(\mathbf{q}^t, y) > \rho E_{\text{aprc}}(\mathbf{q}^t, c^t) \end{cases} \tag{6}$$

where ρ is a tolerance parameter which is slightly larger than 1. The idea is straightforward: if the new prediction does not make any (potential) positive progress on minimizing the appearance energy, then the model will keep the old one.

Formulation and Optimization

The data term can be written as $F(\mathbf{q}) = \mathbf{f}^T\mathbf{q} + \mathbf{g}^T(\mathbf{1} - \mathbf{q})$, where $\mathbf{f}$, $\mathbf{g}$ are foreground and background descriptors, for example $f(x) = -\log(p_{in}(\mathbf{I}(x)))$ and $g(x) = -\log(p_{out}(\mathbf{I}(x)))$ where $p_{in}(*)$ and $p_{out}(*)$ are the color probabilities of object and background, given the current segmentation $\mathbf{q}$. The basic segmenter can be formulated as follows:

$$E_{data}(\mathbf{q}) = F(\mathbf{q}) + \sum_x r_e(x)|\nabla q(x)| \tag{7}$$

where $\mathbf{r}_e = \frac{1}{1+|\nabla \mathbf{I}|}$ is the edge detector. We use kernel density estimation to compute $p_{in}(\mathbf{I})$ and $p_{out}(\mathbf{I})$.

The descriptor $\mathbf{f}$, $\mathbf{g}$ for regions inside and outside the contour change as the segmentation $\mathbf{q}$ changes. The variational segmenter is based on the assumption that parts in the same region share some certain homogeneities, thus it is always exploring possible regions and trying to find the best $\mathbf{q}$ to fit the image data. The result is that it usually is more consistent with image on the boundaries and borders. Moreover, the variational framework allows the segmentation and classification to coordinate with each other.

Our final model can be formulated as:

$$E(\mathbf{q}, c) = E_{data}(\mathbf{q}) + \sum_{i=1}^{L} I(i = c)\mathbf{q}^T(\alpha \mathbf{r}_i - \beta \mathbf{s}_i) \tag{8}$$

Algorithm 1: Optimize Eq.8

Input : Image $\mathbf{I}$, coefficients α, β, ρ, max iteration T and other parameters.
Output: Object mask $\mathbf{q}$, object label c

Compute the generic preference $\mathbf{s}_u$ to generate the initial shape probability $\mathbf{q}^0 \leftarrow \mathbf{s}_u > 0$.
Compute the class-wise preferences $\{\mathbf{s}_i, i = 1 \cdots L\}$.
for $t \leftarrow 1$ **to** T **do**
  Inference : $\mathbf{V} \leftarrow \text{Enc}(\mathbf{q}^{t-1})$
  for $i \leftarrow 1$ **to** L **do**
    Generate reference shapes: $\tilde{\mathbf{q}}_i \leftarrow \text{Dec}(i, \mathbf{V})$
    For each x, Compute $r_i(x) \leftarrow \ln(\frac{1-\tilde{q}_i(x)}{\tilde{q}_i(x)})$
  end
  $y \leftarrow \arg\max_i ||\mathbf{v}_i||$
  Decide to accept or reject new prediction y according to Eq.6 to update c^t
  $\mathbf{r}_q \leftarrow (\mathbf{f} - \mathbf{g}) + \sum_{i=1}^{L} I(i = c^t)(\alpha \mathbf{r}_i - \beta \mathbf{s}_i)$
  Update $\mathbf{q}^t \leftarrow \text{SplitBregman}(\mathbf{I}, \mathbf{q}^{t-1}, \mathbf{r}_q, \mathbf{r}_e)$
end

where $r_i(x) = \ln(\frac{1-\tilde{q}_i(x)}{\tilde{q}_i(x)})$. Here, $(\alpha \mathbf{r}_i - \beta \mathbf{s}_i)$ is the whole expression for the target object, assuming it's of class i, according to the knowledge we have. Performing the joint segmentation and recognition is equivalent to optimizing the energy formulation Eq.2. Since c is the classification depending on $\mathbf{q}$, the minimization of $E(\mathbf{q}, c)$ boils down to optimizing $\mathbf{q}$, which can be done by standard gradient descent methods or other optimization techniques. We employ Split Bregman Method (Goldstein, Bresson, and Osher 2010) to perform the minimization, like (Chen et al. 2013).

A proper initial value $\mathbf{q}^0$ is needed for this optimization. Therefore, besides the class-wise preferences $\{\mathbf{s}_i\}_{i=1,\cdots,L}$, a class-independent preference $\mathbf{s}_u$ is also established, with the regions of objects (regardless the categories) viewed as positive and background negative. It's more like an attention map to tell the model which regions to focus on, thus it is not accurate and detailed enough as the final segmentation result and too coarse for recognition. The initial contour is defined by $\mathbf{q}^0 = \mathbf{s}_u > 0$, and the full procedure is summarised in Algorithm.1. During the optimization, in the first half iterations, we use the soft assignment version of E_{shape} Eq.4, then the hard assignment version Eq.3 for the second half iterations.

Experiments

Signs and Logos

To evaluate our model, we made a dataset including 30 categories of signs and logos commonly found in the wild and on the internet (see Fig.4). These signs and logos can be recognized using specific shape patterns. White regions inside the signs and logos are viewed as the background so that each category differs from others and is recognizable and discriminative to humans and computers. Objects and their corresponding masks are then cropped randomly with different scales ([0.5,0.7] of image size) and locations, generating

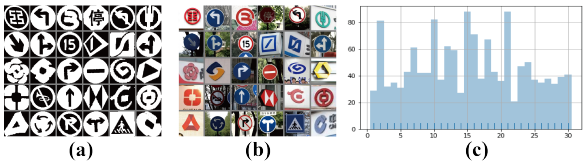

Figure 4: Some information about the dataset. (a) 30 classes of signs and logos. (b) Sample of each class. (c) Numbers of instances of each class.

1436 sample images each with a single sign or logo. Among them, 1080 instances are randomly selected for training, while the other 356 for testing. The 1080 training shapes are augmented using randomly translation, rotation, and projection (within a certain range), generating about 20 thousand shapes in all for the capsule network to capture the deformations and variations.

The structure of the capsule network in this experiment is similar to the one in (Sabour, Frosst, and Hinton 2017): the input is of size 80x80, followed by a standard conv layer with 256 channels of kernel size 11x11 and stride 3, and then 16 types of 16D convolution capsules with kernel size 9x9 and stride 2, finally fully connected with 30 types of 24D capsules, each representing a category (*CategoryCaps*). The decoder is an MLP of layer size [512, 1024, 6400], which takes the output of CategoryCaps as the input.

For comparison, we evaluate the fore- and background segmentation performances by some existing models that employ RBM, such as GLOC(Kae et al. 2013), CHOPPS(Li, Tarlow, and Zemel 2013) and MMBM(Yang, Safar, and Yang 2014), and DeepLab (Chen et al. 2016). We also tested the framework Fig.3 implemented with a standard CNN encoder (and the classification layer), as the **baseline** model, following (Sabour, Frosst, and Hinton 2017).

The **baseline** model begins with three 5x5 convolutional layers of [128,256,256] channels of stride 1, each followed by a 2x2 max-pooling. The last max-pooling layer is followed by two fully connected layers [1080, 690], and then connected to 30 class softmax layer. The classification result (one-hot encoded) is concatenated with the extracted features and fed to the decoder for reconstruction, so that the features used for reconstruction have the same dimension ($24 * 30 = 690 + 30$). The decoder has the same structure. The amounts of parameters of the two networks (proposed and baseline) are ensured to be at the same level.

To establish the preference terms $\mathbf{s}_u$ and $\{\mathbf{s}_j\}$, we first segment the image into superpixels. Like (Yang, Safar, and Yang 2014), dense SIFT, color and contour histograms are computed for each superpixel. Each pixel uses the features of the corresponding superpixel. We set $\alpha = 0.7, \beta = 1.15$ and $\rho = 1.1$. For all the test images, the optimization runs for 100 iterations.

The mean image intersection over-union score (mean IoU) is used to evaluate the segmentation performance, and prediction accuracy for the recognition performance.

Some qualitative results are depicted in Fig.5.

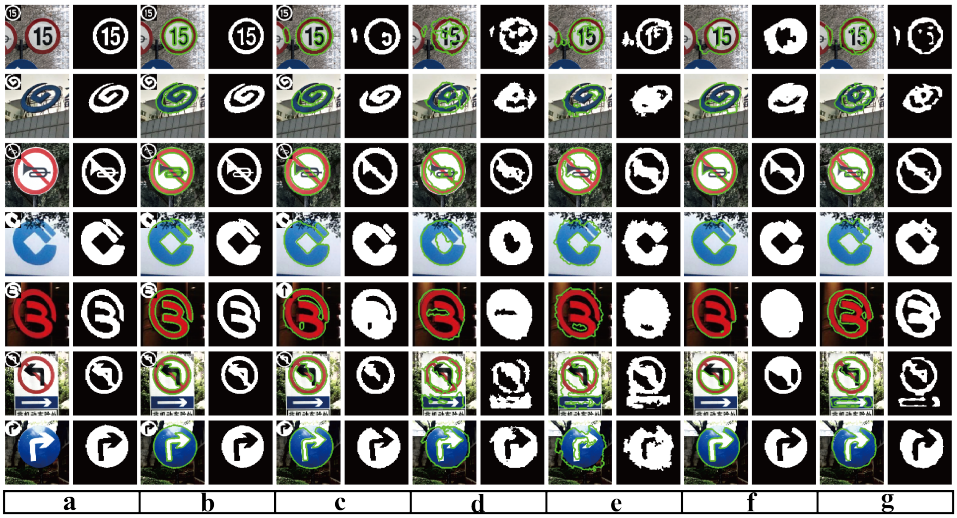

Figure 5: Results of some test samples by different models. Green lines indicate the contours. The recognition results are in the upper left, represented by small category shape images (column (a),(b) and (c)). Best view in color. (a) Test image, the ground truth mask, and the label. (b) Results by the proposed model: the contour and the object mask. (c) Results by the baseline model. (d) Results by CHOPPS. (e) Results by GLOC. (f) Results by MMBM1 w/ GC. (g) Results by DeepLab.

Models	IoU/%	Acc/%
Proposed	**91.00**	**98.01**
Baseline	86.52	90.17
CHOPPS(Li, Tarlow, and Zemel 2013)	69.90	/
GLOC(Kae et al. 2013)	74.12	/
MMBM(Yang, Safar, and Yang 2014)	84.43	/
DeepLab(Chen et al. 2016)	79.29	/

Table 1: Mean IoU and prediction accuracy of different configurations for the signs and logos data.

As is seen, the segmentations obtained by the proposed model with capsules are both accurate and reliable, and thus are correctly recognized by the network meanwhile. For other models, neither the overall segmented object shapes nor their details on the boundaries and local parts are precise as ours. The existing prior-based and RBM-based methods, i.e. CHOPPS/GLOC/MMBM, perform well in structured segmentation, due to the prior shapes learned via RBM. However, only the structure of the segmentation is constrained. In some cases, the object masks can not be recognized, for example, column (d-g) in line (1) in Fig.5. Needless to say, these models are not able to perform recognition. Our model considers both the segmentation and recognition, and based on the whole expression of the object, the proposed model is able to extract the object shapes that are both meaningful and recognizable in most cases. All the metrics are listed in Fig.1, which tells the same result.

For ablation, the results by the same framework but with a standard CNN encoder are as well presented in column (c) of Fig.5 and in Table.1, which are less accurate, in both segmentation and classification. We visualize some of the intermediate generated/reconstructed shapes during segmentation in Fig.6, for the capsule network and the baseline. It is notable that the baseline model is not able to generate a complete object shape or offer a valid reference shape along the evolution of the segmentation. Therefore, its reference shapes are usually less instructional for the segmenter, although it gets the correct recognition sometimes. It is obvious that the baseline model suffers from the interference in the segmentation. This fails it achieving better segmentation and recognition performances. By contrast, in the first sample in Fig.6, the generated shape of the capsule network is at first somehow fuzzy, yet complete. As the segmentation evolves, the reference shape becomes more confident and concrete, thus helping the model to get more accurate segmentation. In the second case, the recognition is incorrect at first, but as the joint tasks proceed, both evolve toward the desired results and produces reliable segmentation and correct recognition.

Hand Gestures

The second experiment is conducted on (Memo, Minto, and Zanuttigh 2015) hand dataset. There are 11 categories of hand gestures acquired with Creative Senz3D camera. Hand gestures typically vary among different persons in different environments and at different times. Each type of hand gestures has a unique shape or silhouette that can be identified, but unlike the signs and logos, it is unnecessary to interpret them as the combinations of simple shapes. However, skins of the human body have the similar color with hands, that is likely to introduce interference during segmentation. Therefore it is not easy to segment out and recognize the hand gesture using pure image data.

We firstly make the segmentation ground truth for the ges-

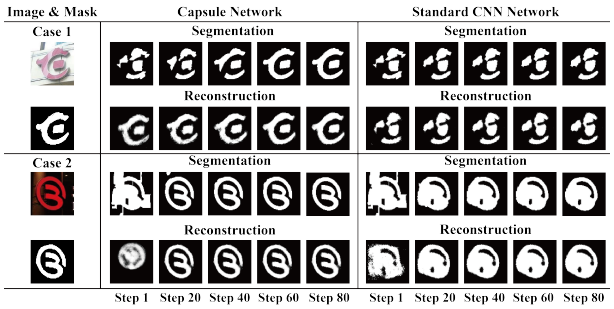

Figure 6: Comparison of reconstructions during segmentation, between the proposed and baseline model. The initial segmentations are the same for the two models.

Models	IoU/%	Acc/%
Proposed	**90.57**	**97.75**
Baseline	88.38	91.00
Original (Memo, Minto, and Zanuttigh 2015)	/	89.90
CHOPPS(Li, Tarlow, and Zemel 2013)	82.30	/
GLOC(Kae et al. 2013)	78.67	/
MMBM(Yang, Safar, and Yang 2014)	87.26	/
DeepLab(Chen et al. 2016)	81.42	/

Table 2: Mean IoU and recognition accuracy of different configurations for the hand gesture data.

tures, palms regions mainly excluding the wrists. The hands are then cropped out with some margins. Of all 1320 images, 920 are chosen randomly for training, and the rest 400 for testing. Hand shapes are augmented with rotation in [-15,15] degrees and small projections randomly, making about 18 thousand hand shapes. The capsule network is basically the same as the one used before, but with 11 category capsules. Again, to make sure the baseline has the similar number of parameters and the features used for reconstruction have the same dimension, this time in the standard CNN **baseline**, the encoder has two fully connected layers ([1024, 256], $24 * 11 \approx 256 + 11$). The decoder remains the same. Here $\alpha = 0.65, \beta = 0.8$. The others remain the same. The authors (Memo, Minto, and Zanuttigh 2015) provided an average accuracy 90% over all the classes for all the 1320 acquisitions.

The statistics of different methods are listed in Table.2. In this dataset, we get similar results. The proposed model with capsule neurons get the best statistics on segmentation over the other models, and achieve better performance on the recognition over the standard CNN model, as is expected (90.57% mean IoU, 97.75% accuracy). Although the **basline** model gets a considerable segmentation performance, its recognition performance is still unsatisfied, compared to the proposed model. The qualitative results are depicted in Fig.7, obtained by the methods listed. Although the faces and arms share the similar color with the hands, our model is able to segment out the hands and meanwhile perform accurate predictions. The segmentation results of the existing

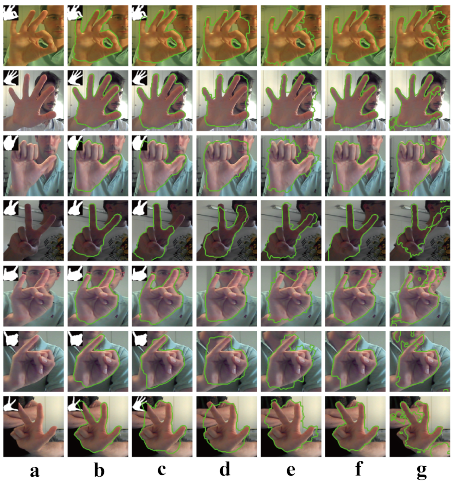

Figure 7: Results of some test samples by different models. (a) Test image, and the label. (b) Results by the proposed model with capsules. (c) Results by the baseline model. (d) Results by CHOPPS. (e) Results by GLOC. (f) Results by MMBM1 with GC. (g) Results by DeepLab.

methods, however, are relatively unsatisfied. Most of them suffer from the interference that comes from the faces and arms. Among them, the segmentation results produced by the MMBM model are the closest to our best and are fairly considerable. Other semantic and deep learning methods obtain a lot of incorrect and coarse segmentation details. As a result, these segmentations hardly can be recognized by humans. All these models need additional modules or branches to tell what the target is.

Conclusion and Discussion

In this paper, we propose a framework that integrates segmentation and recognition, where the two tasks interact and cooperate with each other. Experiments have demonstrated the effectiveness of our coordination model.

Such a framework can also be applied to the objects that can be expressed and identified by their shapes. In fact, objects, such as humans, different animals, different vehicles, alphabet letters, etc., indeed can be recognized using their shapes/shape patterns. Therefore, the framework is not limited to the signs, logos, and hand gestures. The main shortcoming of the framework is the processing speed, since several iterations are required for optimization. Yet, it can be accelerated by parrallel computing methods. We believe that the framework is worth further exploration and extension with other techniques or modules, or combined with other tasks, e.g, scene parsing, to utilize the context to promote these tasks.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant 61471321, the Zhejiang

Science and Technology Plan under Grant 2017C31023, and Artificial Intelligence Research Foundation of Baidu Inc.

References

Chan, T. F., and Vese, L. A. 2001. Active contours without edges. *IEEE Transactions on Image Processing* 10(2):266–277.

Chen, F.; Yu, H.; Hu, R.; and Zeng, X. 2013. Deep learning shape priors for object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1870–1877. IEEE.

Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2016. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *arXiv:1606.00915*.

Chen, F.; Yu, H.; and Hu, R. 2013. Shape sparse representation for joint object classification and segmentation. *IEEE Transactions on Image Processing* 22(3):992–1004.

Cremers, D.; Osher, S. J.; and Soatto, S. 2006. Kernel density estimation and intrinsic alignment for shape priors in level set segmentation. *International Journal of Computer Vision* 69(3):335–351.

Erdil, E.; Yildirim, S.; Cetin, M.; and Tasdizen, T. 2016. Mcmc shape sampling for image segmentation with nonparametric shape priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 411–419. IEEE.

George, D.; Lehrach, W.; Kansky, K.; Lázaro-Gredilla, M.; Laan, C.; Marthi, B.; Lou, X.; Meng, Z.; Liu, Y.; Wang, H.; et al. 2017. A generative vision model that trains with high data efficiency and breaks text-based captchas. *Science* 358(6368):eaag2612.

Goldstein, T.; Bresson, X.; and Osher, S. 2010. Geometric applications of the split bregman method: segmentation and surface reconstruction. *Journal of Scientific Computing* 45(1-3):272–293.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, 2672–2680.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. *arXiv preprint arXiv:1703.06870*.

Kae, A.; Sohn, K.; Lee, H.; and Learned-Miller, E. 2013. Augmenting crfs with boltzmann machine shape priors for image labeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019–2026.

Khan, N.; Algarni, M.; Yezzi, A. J.; and Sundaramoorthi, G. 2015. Shape-tailored local descriptors and their application to segmentation and tracking. In *CVPR*, 3890–3899.

Kihara, Y.; Soloviev, M.; and Chen, T. 2016. In the shadows, shape priors shine: Using occlusion to improve multi-region segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 392–401. IEEE.

Kim, J.; Fisher, J. W.; Yezzi, A.; Cetin, M.; and Willsky, A. S. 2002. Nonparametric methods for image segmentation using information theory and curve evolution. In *Proceedings of the IEEE International Conference on Image Processing*, volume 3, 797–800. IEEE.

Kingma, D. P., and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Larsen, A. B. L.; Sønderby, S. K.; Larochelle, H.; and Winther, O. 2015. Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*.

Lecumberry, F.; Pardo, Á.; and Sapiro, G. 2010. Simultaneous object classification and segmentation with high-order multiple shape models. *IEEE Transactions on Image Processing* 19(3):625–635.

Leventon, M. E.; Grimson, W. E. L.; and Faugeras, O. 2000. Statistical shape influence in geodesic active contours. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, 316–323. IEEE.

Li, C.; Kao, C.-Y.; Gore, J. C.; and Ding, Z. 2007. Implicit active contours driven by local binary fitting energy. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–7. IEEE.

Li, Y.; Tarlow, D.; and Zemel, R. 2013. Exploring compositional high order pattern potentials for structured output learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 49–56.

Long, J.; Shelhamer, E.; and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3431–3440.

Memo, A.; Minto, L.; and Zanuttigh, P. 2015. Exploiting silhouette descriptors and synthetic data for hand gesture recognition. In *Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference*, 15–23. The Eurographics Association.

Mumford, D., and Shah, J. 1989. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics* 42(5):577–685.

Prisacariu, V. A., and Reid, I. 2011. Nonlinear shape manifolds as shape priors in level set segmentation and tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2185–2192. IEEE.

Sabour, S.; Frosst, N.; and Hinton, G. E. 2017. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems*, 3857–3867.

Vecera, S. P., and Farah, M. J. 1997. Is visual image segmentation a bottom-up or an interactive process? *Attention Perception & Psychophysics* 59(8):1280–1296.

Yang, J.; Safar, S.; and Yang, M.-H. 2014. Max-margin boltzmann machines for object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 320–327.