Part Segmentation using 2D Image Surface Normal Estimation

Lyndon Chan
University of Toronto
lyndon.chan@mail.utoronto.ca

***Abstract*—Part segmentation is an important bottom-up vision task that involves the perceptual grouping of lower-level visual stimuli into meaningful structures for higher-level processing informed by semantic information. Currently, there exist many part segmentation methods approaching the problem from either a 2-D Gestalt or medial axis grouping paradigm, or from a 3-D mesh partitioning paradigm. This paper proposes to approach part segmentation through a 2.5-D local surface orientation paradigm, by inferring likely surface orientation from a 2-D image and segmenting object parts with that information. A simple algorithm demonstrating the functionality of the approach is designed in different configurations, and tested on a small set of synthetic and natural images. Experimental results show that the optimal configuration of the proposed method works well for segmenting object parts, provided the object of interest is well-segmented from the background and other irrelevant objects.**

I. Overview

Vision can be conceptualized as the recovery of meaningful, relevant, and useful information about a scene of 3-D objects from a 2-D visual intensity array projected onto a sensor array (the retina for biological images, and an image sensor for digital images). One such vision task, known as part segmentation, concerns the segmenting of the observed image into meaningful parts based on some criteria - research [29] indicates that humans are remarkably capable of extracting meaningful structural patterns even from images of unfamiliar objects, and Wertheimer noted that the human mind tended to apply innate grouping laws to organize visual elements as parts of whole percepts. Among others, he noted that proximal, similar (in shape, colour, or shading), common-fate, continuous, and self-enclosed elements are more likely to be perceived together as grouped organizations, and that these factors interact with one another in a complex way [32].

Subsequent psychophysical research has made further observations about human visual perception. [15] showed that human texture discrimination is the result of local fundamental features called textons. [6] showed that humans are more perceptually sensitive on a 2-D contour closure continuum. [17] theorized that the human mind extracts medial-axis "skeletons" as intermediate-level representations of visual objects. And [11] argued that the human visual system divides perceived surfaces into parts along contours of concave discontinuity.

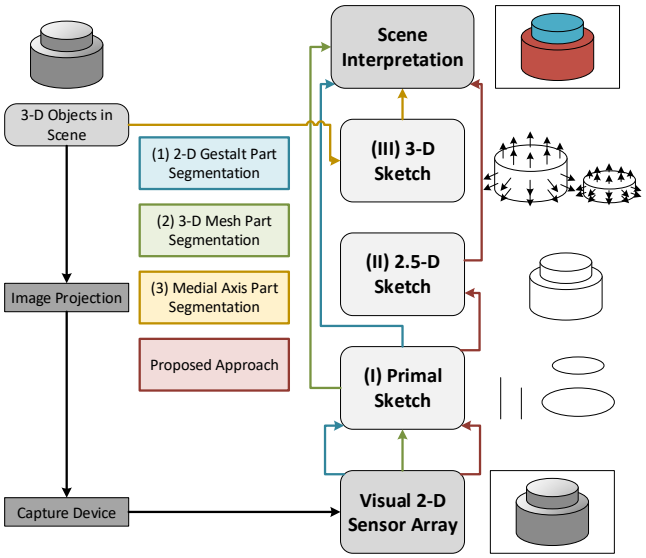

Fig. 1. Four possible approaches to part segmentation, considered under Marr's three-stage formulation of the vision problem

These findings on human vision suggest that humans might be decomposing highly complex scene images using a grammar of visual primitives, and then group them into more manageable parts which make up whole objects. This is helpful for computational vision because, if replicated in a computational framework, it potentially provides a compact but meaningful shape representation for extracting relevant information about the 3-D scene

from 2-D digital images.

As mentioned above, the part segmentation task aims to divide an image into meaningful parts, usually by some semantic, appearance, or shape criteria. In doing so, an imaged object can be represented as an articulated arrangement of simpler geometric components in order to facilitate subsequent recognition or other more abstracted visual processes informed by higher-level knowledge and purpose [21]. Since these parts are more invariant than individual pixels over viewing position and conditions, solving the part segmentation problem permits robust object detection even from a novel viewpoint or subjected to a similarity transform, assuming that part decomposition is still possible.

II. Related Work

There are many possible approaches to solve the part segmentation problem; three existing approaches are detailed in this section.

(1) 2-D Gestalt Part Segmentation
These part segmentation algorithms assume that imaged object parts comprise of visual primitives (e.g. zero-crossings, edges, bars, blobs) which are grouped together by Gestalt grouping laws. A particularly widespread approach is to model object parts as self-enclosed regions of the image bounded by maximally-convex non-accidental 2-D contours. Graph theoretic frameworks are used by [18] to maximize superpixel contour closure and compactness, while [23] groups image "edgels" by continuity and proximity.

(2) 3-D Mesh Part Segmentation
These part segmentation algorithms assume that object parts are regions bounded (in three-dimensions) by maximally-convex surfaces, usually by applying a graph-theoretic approach like Normalized Graph Cuts [24] to 3-D mesh data. [19] use spectral clustering, [10] use randomized cuts, and [16] use fuzzy clustering and mean cuts.

(3) Medial Axis Part Segmentation
These part segmentation algorithms decompose closed 2-D shapes into sets of symmetric skeletal parts centred around linear or curvilinear axes. [2] proposed the medial axis transform, [25] used shock graphs (graphs of singularities in inward silhouette evolution), and [7] approached skeletonization through a Bayesian probabilistic framework.

III. Method

In this section, I give an overview of the proposed two-stage method for part segmentation using estimated local surface normal information as an intermediate shape representation. Then, I will describe the different aspects of the part segmentation problem addressed by each stage, and the functionality of the considered methods for each stage.

A. Overview

As mentioned earlier, there are three main approaches to part segmentation: (1) the 2-D Gestalt approach, (2) the 3-D Mesh approach, and (3) the Medial Axis approach. The main weakness of the first approach is its reliance on detecting the visual primitives first in order to infer the underlying shape, which is susceptible to appearance variations (e.g. lighting, texture). And by strictly requiring an enclosed 3-D mesh to segment, the second approach is unsuitable for performing part segmentation on a 2-D image, short of including an initial 3-D mesh estimation step. The third approach, similarly to the first approach, is reliant on accurate contour detection, which can be difficult for highly textured images.

These approaches can be understood as selectively incorporating various levels of information from Marr's three-stage formulation of the vision problem [22]. Here, vision is seen as proceeding in sequence from a two-dimensional visual array to: (1) a **primal sketch** comprising of primitive scene components (e.g. zero-crossings, blobs, edges, bars), (2) a **2.5-D sketch** comprising of viewer-centred 2-D planar projections of local 3-D surface scene elements, and finally to (3) a **3-D sketch** of hierarchical models of 3-D volumetric models, which when incorporating top-down information can facilitate the extraction of useful information about the scene underlying the original visual array. Interpreted through Marr's framework, the first and third approaches use primal sketch information to describe the 3-D sketch, and the second approach describes the 3-D sketch straight from 3-D information known *a priori* (see the blue, green, and yellow arrows in Figure 1 for the conceptualization of approaches (1), (2), and (3) respectively under Marr's three-stage formulation).

Hence, I propose to approach the part segmentation problem by extracting the surface normal information as an intermediate 2.5-D sketch representation of shape to bridge the primal and 3-D sketch stages (see the red arrows in Figure 1 for the conceptualization of the proposed approach under Marr's three-stage formulation). I hypothesize that this approach provides the algorithm with richer shape representation and enables more accurate part segmentation. The proposed approach

(visualized in Figure 2) divides the problem into two stages: (1) estimating local surface normal information from a 2-D image, then (2) performing part segmentation on the local surface normal information.

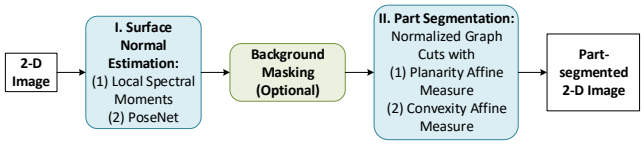

Fig. 2. The proposed approach to part segmentation, consisting of a surface normal estimation stage (with two considered methods), followed by a part segmentation stage (also with two considered methods)

B. Stage 1: Surface Normal Estimation

Overview

In computational vision, the problem of 3-D shape recovery from one or more 2-D images of it is inherently ambiguous and technically ill-posed. Hence, it is usually approached by modelling some aspect of the relationship between physical surfaces and images of them. This paper will focus on techniques that use only single 2-D images (such as texture and shading), as opposed to techniques that use multiple images (such as stereo and motion).

The Shape from Texture (SFT) approach is motivated by Gibson's observation [9] that a gradual local variation in the density of a uniform pattern of repeated texture elements (known as "texture gradient") in an observed brightness image indicates the direction and degree of projection foreshortening. By assuming either texture homogeneity or isotropy, the SFT approach models a frontal texture and aims to recover the surface normal at each pixel in the image. The Shape from Shading (SFS) approach, on the other hand, models the observed brightness image as a function of the surface normal and the light source direction [12], and attempts to recover the surface normal by recovering the light source and modelling the likely shape responsible for the observed image [34]. Another technique, used more widely in recent years, is to take a data-driven approach to estimating surface normals, such as those proposed by [8], [5], [33], [35], and [1].

In this section, I will detail two proposed methods for surface normal estimation. The first is a hand-crafted homogeneous-assumption Shape from Texture technique modelled after the Super and Bovik's local spectral moment algorithm [28], and the second is the data-driven PoseNet convolutional neural network by Bansal et al. [1]

Method 1: Using Local Spatial Moments

Super and Bovik [28] describe a non-feature-based method for estimating the surface normal at each image pixel using the moments of local spatial-frequency spectra. The authors base their technique on the observation that the projection of local spectra and spectral moments of any surface reflectance patterns ("surface markings") oriented in 3-D space correspondingly transforms those surface patterns' local spatial-frequency spectra as well. They propose using Gabor filters to sample the local spatial-frequency spectra in a computationally-efficient manner, and then compute the contrast-normalized canonical moments (structure tensors) of the sampled spatial-frequency spectra. They define the effect of orthographic projection on the local spatial frequency (and hence, their canonical moments), and then prove that computing the canonical moments of two image pixels is sufficient to calculate the relative slant between their local surfaces, and that two solutions for the tilt angle can be calculated from each slant angle estimate up to a 180-degree ambiguity. Since the slant estimation is relative, a frontal slant must be known (or computed) beforehand, and the surface normals of all other points in the image can be estimated from it.

In this paper, I replicate Super and Bovik's method, but replace the Gabor filters with MaxPol filters introduced in [13] [14], which fulfill the maximal-flatness criterion and provide better sampling of the local spatial-frequency spectrum. Initial tests indicate that fewer MaxPol filters are consequently able to perform better local spatial-frequency sampling than Gabor filters.

Method 2: Using A Convolutional Neural Network

Bansal et al. [1] describe a skip-network fully-convolutional neural network for pose and style prediction, consisting of six convolutional layers initialized with the pre-trained VGG-16 network weights [27] from the 5-layer (of 13) subset $\{1_1, 2_2, 3_3, 4_3, 5_3, 7\}$, followed by two fully-connected layers (out of 3) from VGG-16 (*fc-6* and *fc-7*), converted into convolutional layers in the manner of [20]. The surface normal estimation network, named "PoseNet", is then trained on the surface normals computed from the Kinect depth channel of the raw video frames of the NYU Depth v2 dataset [26]. This information is computed by Ladicky et al. [33] for the validation and test sets, and using the approach of Wang

et al. [30] for the training set. The authors perform an ablative analysis on the performance of different network architectures and show that the proposed 5-layer design performs best in all six surface normal estimation criteria introduced by Fouhey et al. [8].

C. Stage 2: Part Segmentation

Overview

Given the surface normal information estimated by the first stage, the second stage of the proposed method is to partition the image of a single object into its meaningful parts. In this paper, I take a graph-theoretic formulation of the part segmentation problem proposed by Shi and Malik [24]. Known as the Normalized Graph Cut, this method formulates the image segmentation problem as the generalized eigenvalue problem of partitioning a graph of pixels in the 2-D image and minimizes the normalized cut criterion in order to maximize the image partition's goodness of fit. As proposed by Shi and Malik, the grouping algorithm proceeds for a 2-D image I as follows:

1) Set up a weighted graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ consisting of a single node per pixel and an edge connecting each pair of neighbouring pixels with an associated weight computed by the affinity measure (which reflects the likelihood that the two pixels belong to the same grouping)
2) Solve $(\mathbf{D} - \mathbf{W})\mathbf{x} = \lambda\mathbf{D}\mathbf{x}$ for eigenvectors with the smallest eigenvalues
3) Use the eigenvector with the second smallest eigenvalue to bipartition the graph
4) Recursively repartition the segmented parts if necessary

Thus, to take the bipartite case as an example, a graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ is partitioned into two disjoint sets $A, B, A \cup B = V, A \cap B = \emptyset$ such that the normalized cut ("Ncut") of the graph is minimized:

$$
\begin{aligned}
Ncut(A,B) &= \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}, \\
cut(A,B) &= \sum_{w \in A, v \in B} w(u,v), \\
assoc(A,V) &= \sum_{u \in A, t \in V} w(u,t), \\
assoc(B,V) &= \sum_{v \in B, t \in V} w(v,t).
\end{aligned}
$$

For both of the considered part segmentation methods, the Normalized Graph Cuts MATLAB code provided by Shi [1] was implemented with the following two custom affinity measures.

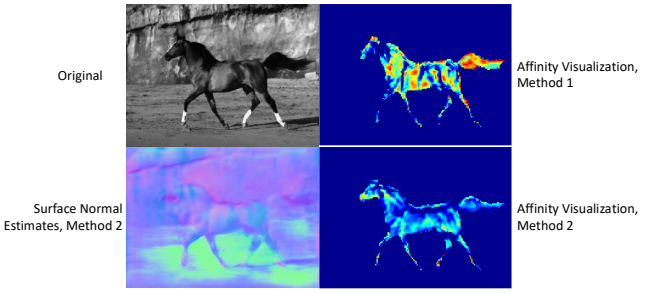

Fig. 3. Visualization of the two considered affinity measures on the `horse001` image from the Weizmann Horses dataset, using MATLAB's `jet` colour mapping, adjusted to enhance details. The surface normal image is coloured with positive x-direction (left) in red, positive y-direction (up) in green, and positive z-direction (toward viewer) in blue. This colour coding for surface normals is used throughout the rest of this paper.

Method 1: Using Planar-Affinity Normalized Cuts

In the first part segmentation method, I define the normalized cut affinity measure $w(\mathbf{u}, \mathbf{v})$ between two 8-neighbouring surface unit normals $\mathbf{u} = (u_x, u_y, u_z), \mathbf{v} = (v_x, v_y, v_z)$ to be the negative of the exterior dihedral angle between them:

$$
w(\mathbf{u}, \mathbf{v}) = -\cos^{-1}(|\mathbf{u} \cdot \mathbf{v}|).
$$

This affinity measure, once scaled to the range $[0, 1]$, measures the similarity of surface orientation between $\mathbf{u}$ and $\mathbf{v}$, which will produce graph-cut parts which are maximally planar within themselves and maximally non-planar at their boundaries. This can be considered the 2.5-D analogue of part segmentation by 2-D feature collinearity or co-curvilinearity used by other methods. See Figure 3, top-right, for a visualization of the local affinity measure (i.e. the sum of each pixel's 8-neighbouring affinities) using the first method.

Method 2: Using Convex-Affinity Normalized Cuts

In the second part segmentation method, I define the normalized cut affinity measure $w(\mathbf{u}, \mathbf{v})$ between two 8-neighbouring surface unit normals $\mathbf{u} = (u_x, u_y, u_z), \mathbf{v} = (v_x, v_y, v_z)$ to be the sum of the opposite direction projections of $\mathbf{u}$ and $\mathbf{v}$ onto the image plane:

[1] http://www.cis.upenn.edu/~jshi/software/

$$w(\mathbf{u},\mathbf{v}) = \begin{cases} u_y - v_y, & \text{if } \mathbf{u} \text{ above } \mathbf{v} \\ -u_x + v_x, & \text{if } \mathbf{u} \text{ left of } \mathbf{v} \\ \frac{-u_x+v_x+u_y-v_y}{\sqrt{2}}, & \text{if } \mathbf{u} \text{ top-left of } \mathbf{v} \\ \frac{-u_x+v_x-u_y+v_y}{\sqrt{2}}, & \text{if } \mathbf{u} \text{ top-right of } \mathbf{v} \end{cases}$$

This affinity measure, once scaled to the range $[0, 1]$, measures the degree of convexity between $\mathbf{u}$ and $\mathbf{v}$, which will produce graph-cut parts that are maximally convex within themselves and maximally concave at their boundaries. This can be considered the 2.5-D analogue of part segmentation by 2-D contour or 3-D mesh convexity used by other methods. See Figure 3, bottom-right, for a visualization of the local affinity measure (i.e. the sum of each pixel's 8-neighbouring affinities) using the second method.

IV. Experiments

A. Stage 1: Local Surface Orientation Estimation

Experiment 1: Synthetic Textured Spheres

First, the two considered methods of local surface orientation estimation were tested on five simple synthetically-generated images of a texture patch projected onto a spherical mesh with perspective projection and Lambertian reflectance. Two are synthetically generated isotropic textures (`checker` and `dot`), while three are natural textures from the Brodatz texture set [31] ("D9" for `grass`, "D16" for `weave`, and "D68" for `wood`).

Figure 4 displays the results; the first method (using local spectral moments) gives a reasonable estimate for the synthetically-generated isotropic textures, but fails to give any reasonable estimate for the natural textures (which may be anisotropic), whereas the second method (using PoseNet) performs much more reasonably. From now on, only the second method (using PoseNet) of local surface orientation estimation will be used for part segmentation.

B. Stage 2: Part Segmentation

Experiment 2a: Weizmann Horses (with Background Masking)

To evaluate part segmentation performance, I first apply both considered methods (with a pre-set graph cut cluster count of 5) on the Weizmann Horses dataset [3], which consists of 328 grayscale images of horses collected from the Internet in different poses and activities, but generally viewed in profile facing left and with minimal or no occlusion. Also, the dataset provides a ground-truth figure-ground segmentation, so I mask out the background from the estimated surface normals before performing the part segmentation. Solving part segmentation on the Weizmann Horses dataset is relatively simple, since surface details are clearly visible and occlusion is minimal, so I use it to confirm reasonable part segmentation performance.

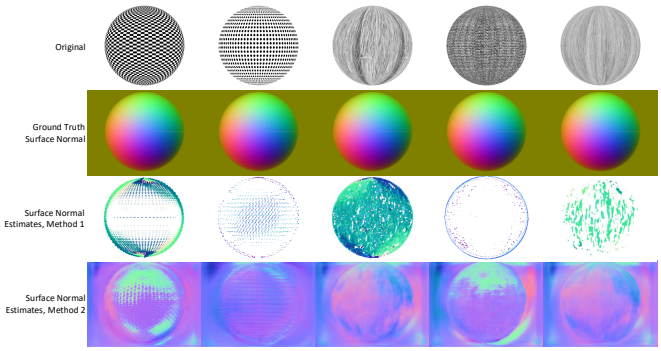

Fig. 4. The estimated local surface normals on the synthetic textured sphere images `checker`, `dot`, `grass`, `weave`, and `wood`

Figure 5 shows the successful part segmentation cases on a subset of the Weizmann Horses dataset. Two observations may be made from this:

1) The second method (using surface convexity) consistently segments the horse into its head, chest and fore-legs, abdomen, and rear-end and hind-legs, whereas the first method (using surface planarity) has a less consistent segmentation; and
2) The second method tends to correspond better to the horse's anatomy (e.g. it segments the highly-convex shoulder region separately from the abdomen, but the first method separates the shoulder into three parts).

Fig. 5. The successful part segmentation results on horses 5, 30, 105, 102, and 169 of the Weizmann Horses dataset with background masking

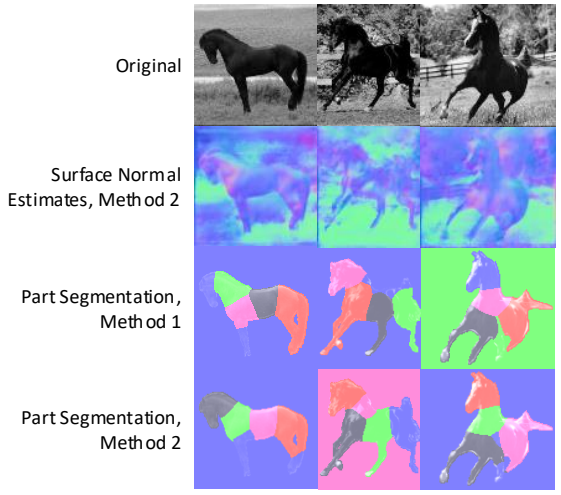

Fig. 6. The failed part segmentation results on horses 125, 215, and 265 of the Weizmann Horses dataset with background masking

Figure 6, meanwhile, shows the failure modes of part segmentation, and both considered methods can be seen to mis-segment certain parts of the horse with the background in cases where the surface normal estimation is too noisy (e.g. at the back of horse 215). In general, the second method of surface normal estimation is accurate enough for reasonable part segmentation, but in cases where the estimation is poor, part segmentation accuracy deteriorates accordingly.

Experiment 2b: PASCAL VOC 2010 Part Database (with Background Masking)

To evaluate part segmentation on a more challenging set of natural images, I apply the two considered methods on the training/validation set of the PASCAL VOC 2010 Part Database [4], which consists of 10,103 images of objects in cluttered scenes segmented into their constituent semantic parts (e.g. the tail of a cat, or the engine of an aeroplane). Since a ground-truth part segmentation is provided with the dataset (along with a ground-truth figure-ground segmentation), I pre-set the graph cut cluster count to equal the number of ground-truth parts provided and again mask out the background from the estimated surface normals.

Figure 7 shows the successful part segmentations on a subset of the VOC dataset: both methods perform surprisingly well for certain images (e.g. the aeroplane in 2008_000033, the cat in 2008_000960, the horse in 2008_000602, the motorbike in 2008_001525, and the person in 2008_000144), although they tend to over-segment when the ground-truth segmentation contains numerous parts. It can also be remarked that the second method tends to out-perform the first method, since surface convexity seems to correspond better to semantic part boundaries than surface planarity (e.g. for the motorbike in 2008_001525).

Fig. 7. The successful part segmentation results on images 2008_000033, 2008_000960, 2008_002045, 2008_000973, 2008_000602, 2008_001525, and 2008_000144 of the PASCAL VOC 2010 Part dataset with background masking

Figure 8, on the other hand, shows the failure modes of part segmentation. Two observations may be made from these:

1) In these images, the ground-truth consists of numerous semantic parts of varying relative sizes. Since normalized graph cuts normalizes is normalized to the number of inter-cut connections, both methods 1 and 2 tend to produce similarly-sized parts.
2) It can be seen from the surface normal estimates that the smaller semantic parts (such as the eyes in 2008_000096 and 2008_000215) are not clearly defined in the surface normal estimation, and this limited estimation resolution is inevitably another cause of the failure in part segmentation.

Experiment 2c: PASCAL VOC 2010 Part Database (without Background Masking)

The proposed two-stage part segmentation algorithm uses shape as a cue for part segmentation after the underlying surface normal has been estimated from the image appearance. As such, it will be susceptible to mis-segmentation when the background is not properly masked, or when other cues are more important for part segmentation than surface shape. Hence, to test the effect of background masking on the part segmentation,

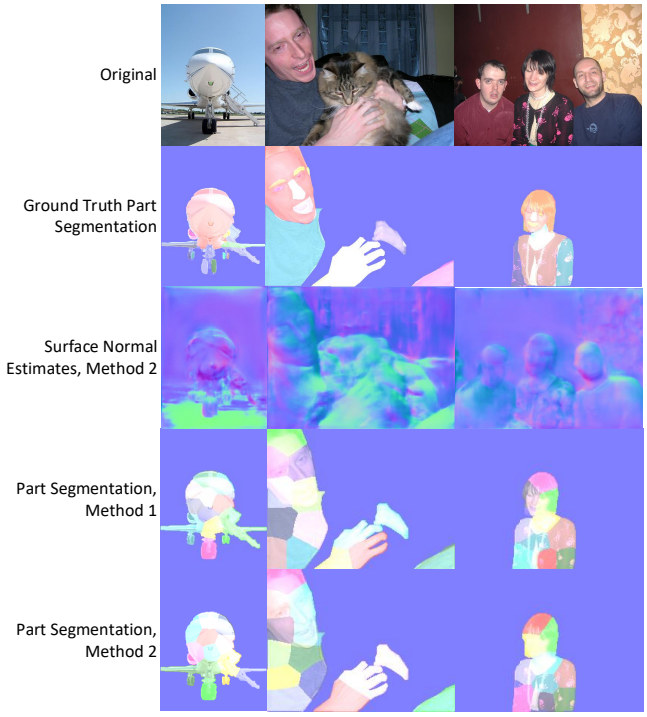

Fig. 8. The failed part segmentation results on images `2008_000064`, `2008_000096`, and `2008_000215` of the PASCAL VOC 2010 Part dataset with background masking

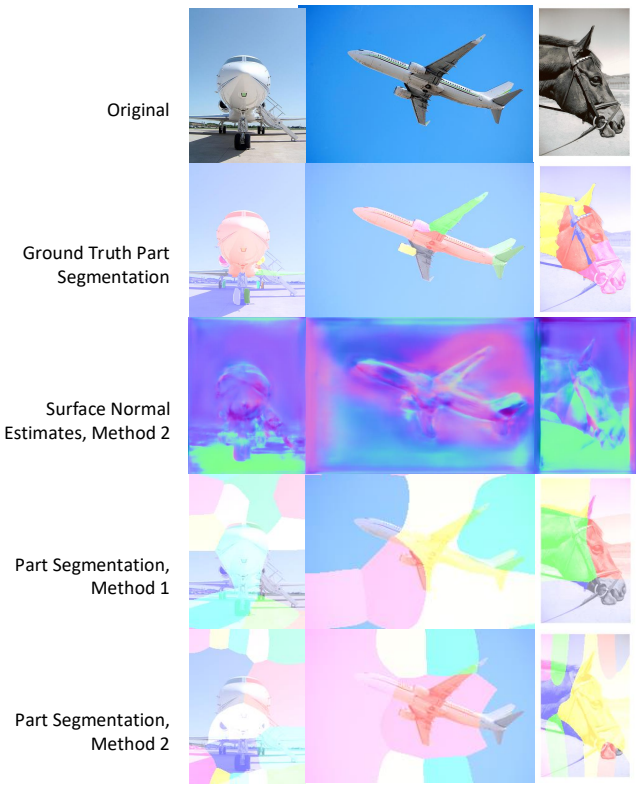

Fig. 9. The part segmentation results on images `2008_000064`, `2008_000033`, and `2008_001525` of the PASCAL VOC 2010 Part dataset without background masking

I conducted a final test on the PASCAL VOC 2010 Part Database, but without first masking out the background.

Figure 9 shows that the part segmentation is wildly off the mark when the background is not masked out. There are likely several reasons for this:

1) The surface normal estimation already has spurious artifacts in the background (e.g. `2008_000033` in the middle), and
2) There is often no clear separation between the object and the background in the estimated surface normals, which leads foreground regions being grouped with background regions.

These results strongly suggest that good relevant object segmentation (and by extension, good foreground-background segmentation) is a strict pre-requisite for the proposed part segmentation algorithm to work properly.

V. Discussion / Future Directions

To conclude, experimental results support the hypothesis that a two-stage part segmentation pipeline consisting of a surface normal estimation stage followed by a graph-theoretic part segmentation stage is a feasible approach to part segmentation. By incorporating surface normal estimates as intermediate shape cues from the image (corresponding to the 2.5-D sketch formulated by Marr), the proposed approach is able to provide reasonable part segmentations. Preliminary tests suggest that the best-performing configuration consists of PoseNet surface normal estimation followed by surface-convexity normalized graph cut segmentation.

Whereas the first proposed method for surface normal estimation proved to be insufficiently robust across different images, the PoseNet convolutional neural network approach was demonstrably accurate enough for part segmentation by normalized graph cuts. Of the two graph-cut affinity measures considered, the second (using surface convexity) was superior to the first (using surface planarity). This can be considered the 2.5-D equivalent of using 2-D contour convexity as a segmentation cue, which is used by other methods. My tests also show that the considered configurations only perform reasonably well when the foreground (or the object of interest) is adequately segmented from the background (or any distractor object).

The proposed method has been shown to work well in certain cases, but its weaknesses suggest some fu-

ture directions for improvement: (1) improve part segmentation by incorporating more bottom-up information (e.g. appearance cues) or more top-down information (e.g. semantic information), (2) implement automatic object of interest segmentation in case such segmentation is not known *a priori*, and (3) address the challenge of segmenting different-sized semantic parts by using a non-graph-theoretic segmentation method or a graph-theoretic segmentation method which accounts for differently-sized clusters.

References

[1] A. Bansal, B. Russell, and A. Gupta. Marr revisited: 2d-3d alignment via surface normal prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5965–5974, 2016. 3

[2] H. Blum. A Transformation for Extracting New Descriptors of Shape. In W. Wathen-Dunn, editor, *Models for the Perception of Speech and Visual Form*, pages 362–380. MIT Press, Cambridge, 1967. 2

[3] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *European conference on computer vision*, pages 109–122. Springer, 2002. 5

[4] X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1971–1978, 2014. 6

[5] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in neural information processing systems*, pages 2366–2374, 2014. 3

[6] J. Elder and S. Zucker. The effect of contour closure on the rapid discrimination of two-dimensional shapes. *Vision Research*, 33(7):981–991, 1993. 1

[7] J. Feldman and M. Singh. Bayesian estimation of the shape skeleton. *Proceedings of the National Academy of Sciences*, 103(47):18014–18019, 2006. 2

[8] D. F. Fouhey, A. Gupta, and M. Hebert. Data-driven 3d primitives for single image understanding. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3392–3399. IEEE, 2013. 3, 4

[9] J. J. Gibson. The perception of the visual world. 1950. 3

[10] A. Golovinskiy and T. Funkhouser. Randomized cuts for 3d mesh analysis. In *ACM transactions on graphics (TOG)*, volume 27, page 145. ACM, 2008. 2

[11] D. D. Hoffman and W. A. Richards. Parts of recognition. *Cognition*, 18(1-3):65–96, 1984. 1

[12] B. K. Horn. Shape from shading: A method for obtaining the shape of a smooth opaque object from one view. 1970. 3

[13] M. S. Hosseini and K. N. Plataniotis. Derivative kernels: Numerics and applications. *IEEE Transactions on Image Processing*, 26(10):4596–4611, 2017. 3

[14] M. S. Hosseini and K. N. Plataniotis. Finite differences in forward and inverse imaging problems: Maxpol design. *SIAM Journal on Imaging Sciences*, 10(4):1963–1996, 2017. 3

[15] B. Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 290(5802):91, 1981. 1

[16] S. Katz and A. Tal. *Hierarchical mesh decomposition using fuzzy clustering and cuts*, volume 22. ACM, 2003. 2

[17] I. Kovacs and B. Julesz. Perceptual sensitivity maps within globally defined visual shapes. *Nature*, 370(6491):644, 1994. 1

[18] A. Levinshtein, C. Sminchisescu, and S. Dickinson. Optimal image and video closure by superpixel grouping. *International journal of computer vision*, 100(1):99–119, 2012. 2

[19] R. Liu and H. Zhang. Segmentation of 3d meshes through spectral clustering. In *Computer Graphics and Applications, 2004. PG 2004. Proceedings. 12th Pacific Conference on*, pages 298–305. IEEE, 2004. 2

[20] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional models for semantic segmentation. In *CVPR*, volume 3, page 4, 2015. 3

[21] D. Marr. Early processing of visual information. *Phil. Trans. R. Soc. Lond. B*, 275(942):483–519, 1976. 2

[22] D. Marr. Vision: a computational investigation into the human representation and processing of visual information. w. h. *WH San Francisco: Freeman and Company*, 1982. 2

[23] Y. Qi, Y.-Z. Song, T. Xiang, H. Zhang, T. Hospedales, Y. Li, and J. Guo. Making better use of edges via perceptual grouping. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 1856–1865. IEEE, 2015. 2

[24] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000. 2, 4

[25] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, and S. W. Zucker. Shock graphs and shape matching. *International Journal of Computer Vision*, 35(1):13–32, 1999. 2

[26] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012. 3

[27] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3

[28] B. J. Super and A. C. Bovik. Shape from texture using local spectral moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(4):333–343, 1995. 3

[29] J. M. Tenenbaum and A. Witkin. On the role of structure in vision. *Human and machine vision*, pages 481–543, 1983. 1

[30] X. Wang, D. Fouhey, and A. Gupta. Designing deep networks for surface normal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 539–547, 2015. 4

[31] A. G. Weber. The usc-sipi image database version 5. *USC-SIPI Report*, 315:1–24, 1997. 5

[32] M. Wertheimer. Laws of organization in perceptual forms. 1938. 1

[33] B. Zeisl, M. Pollefeys, et al. Discriminatively trained dense surface normal estimation. In *European conference on computer vision*, pages 468–484. Springer, 2014. 3

[34] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah. Shape-from-shading: a survey. *IEEE transactions on pattern analysis and machine intelligence*, 21(8):690–706, 1999. 3

[35] Y. Zhang, S. Song, E. Yumer, M. Savva, J.-Y. Lee, H. Jin, and T. Funkhouser. Physically-based rendering for indoor scene understanding using convolutional neural networks. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3