Skeleton-based Interactive Object Co-Part Segmentation

by

Harsh Vardhan Bhadauriya

Under the supervision of

Dr. Koteswar Rao Jerripothula

Indraprastha Institute Of Information Technology Delhi

May, 2023

Skeleton-based Interactive Object Co-Part Segmentation

by

Harsh Vardhan Bhadauriya

Submitted

in partial fulfillment of the requirements for the degree of

Master of Technology

to

Indraprastha Institute Of Information Technology Delhi

May, 2023

Certificate

This is to certify that the thesis titled **"Skeleton-based Interactive Object Co-Part Segmentation"** being submitted by **Harsh Vardhan Bhadauriya** to the Indraprastha Institute of Information Technology Delhi, for the award of the Master of Technology, is an original research work carried out by him under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

May, 2023

Dr. Koteswar Rao Jerripothula
Assistant Professor
Department of Computer Science & Engineering
Indraprastha Institute of Information Technology Delhi
New Delhi, 110 020

ACKNOWLEDGEMENTS

I want to thank my thesis advisor, Dr. Koteswar Rao Jerripothula, for his constant support and supervision. I am immensely grateful to him for providing me with his valuable guidance and insights throughout the course of the thesis. I want to express my gratitude to my family for providing me with unfailing support and continuous encouragement throughout the research and writing of this thesis. If it were not for my sister's support, this journey would not have been possible. Thank you for everything.

PUBLICATIONS

This is a list of submitted work during my masters thesis.

1. **Harsh Vardhan Bhadauriya** and Koteswar Rao Jerripothula, "Skeleton-based Interactive Object Co-Part Segmentation", submitted to Journal of Visual Communication and Image Representation (JVCI) [submitted]

ABSTRACT

Skeleton-based Interactive Object Co-Part Segmentation
Harsh Vardhan Bhadauriya

Object co-part segmentation, which involves segmenting shared objects into meaningful parts in a group of images, is a challenging joint-processing task. Although fully unsupervised deep learning algorithms exist for this task, the resultant parts often lack semantic meaning. This is because these algorithms use latent space to separate the parts, which may not necessarily correspond to meaningful parts as perceived by humans. Additionally, the number of parts required by these algorithms is difficult to pre-determine due to pose and size variations shared objects may exhibit across images, making human interaction necessary. While some interactive methods exist, none of them have explored the use of skeletons, which provide an object structure that can be leveraged to generate meaningful parts. Our proposed approach addresses this gap by presenting a skeleton-based interactive co-part segmentation framework that draws benefits from both unsupervised deep learning and human interaction. The framework employs the correspondence capabilities offered by deep learning counterparts and utilizes skeletons to generate meaningful parts. Experiments on Pascal-Part dataset demonstrate that our proposed framework outperforms existing interactive co-part segmentation methods in terms of segmentation accuracy and meaningfulness of parts.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

A significant aspect of computer vision is analyzing the objects surrounding us, such as animals, humans, vehicles, and others. We want to model machines to perceive the world the way we do. How does a child learn to understand what makes a cat different from a human? They see each object as a sum of their parts. A cat consists of a tail, four legs, etc., while a human has two legs and hands. Their body shape is different from that of a cat. We look at all these fine details and differentiate between uncommon objects based on the sum of their parts. The ability to look at any object and figure out what its different meaningful parts are, where they start, and where they end opens up a whole new usefulness of computer vision algorithms authored for studying objects.

In recent times, noteworthy progress has been made in object-level recognition tasks such as object detection, object segmentation, and semantic segmentation. With such advancements, more precise understanding of an object's composition has been the topic of research. Now, we have models which can easily perform semantic segmentation of a scenery with high accuracy and the quality of object silhouettes are exemplary. This allows us to deep dive into what is the object silhouette consists of. How can we draw lines/boundary inside the object region such that it divides the original object into meaningful/semantic parts. If we can understand what parts an object is composed of and where they are located, it can help solve and refine a variety of visual tasks such as fine-grained recognition [18, 15], pose estimation [19], person re-identification [20, 21], and action recognition [22].

1.1 Problem Statement

Part segmentation is extracting semantic parts of objects in an image. It is the pixel-level semantic labeling of an object into its various parts. The number of semantic parts may differ across objects. The general problem of semantic part segmentation is that if given an image with at least one object, we can identify the different objects with part-level details of each object. For example, in an image consisting of a man sitting on a bike, ideally, we should be able to decompose the bike into tires, headlight, seat, handle, gas pump, etc., and the man into usual human body parts such as head, leg, torso, hand, etc. If we look closely into this problem statement, it is very vague in nature. A model/algorithm does not look at an object and understand what makes it work and how it can be divided into different and meaningful parts. It has to learn similar information during the training phase, and after that, can make such predictions.

The semantic part generation problem depends on what kind and how much information we are willing to share with the network/algorithm. Based on that, we can formulate the problem statement. For example, a supervised approach trained on labeled ground truth can predict an unseen object point with all the possible parts it saw during training. However, this is not possible for unsupervised approaches. Unsupervised methods do not interact with ground truth, so it does not know how many meaningful parts are possible in each object. This information must be passed to the method, i.e., the number of expected parts should be pre-defined. The unsupervised approach will try to generate the set number of expected parts each time.

The problem statement's specificity will keep changing depending on what learning approach is taken and what is input. However, the end result will always be object decomposition in meaningful parts. In this work, we explore ideas at the intersection of unsupervised(no ground truth availability), weakly supervised(weak supervision in the form of object skeleton), and interactive approach(in the form of the user interaction).

1.2 Motivation

We will try to understand what is lacking in current part segmentation methods which lead to pursue this work. The supervised approaches [23, 24, 5, 19, 25] rely on labeled ground truth to train the model or make the algorithm learn. In such cases, the model needs to see the examples from the classes that it is supposed to predict. The supervised approach predictions are good but reliance on annotated data is a major drawback. Unlike object segmentation, part segmentation annotation is a manually extensive task. It is not possible to keep on manually annotating thousands of training examples for each class. The method also performs poor on the unseen classes.

In order to overcome the drawback of the supervised approaches, research explored unsupervised method [17, 7, 26, 6, 27] for part segmentation. These approaches do not require annotated data however the "semantic" nature of the segmentation is lost. The unsupervised approaches are based on self-supervised learning or devising the loss functions such that they mimic the part generation constraints. The number of parts to be extracted are predefined. The major drawback is that if an image of cat has four part and another has five parts, the method will always segment into the number of parts it was trained for(lets say four). These methods are unable to understand and work on the dynamic nature of different poses of an object. Another drawback is that the parts produced are not meaningful in nature. They do not generally correspond to what we humans will call a part because these methods are based on finding similar patterns across images of the same object, and often these similarities across images do not correspond to a semantic part. Also, we need to train a new model each time we want to introduce a new class or change the number of parts to extract.

There are other approaches such as weakly supervised approach [28, 15, 11, 8, 29] and interactive approach [9, 30] which look at reducing the amount of training data required or introducing user interaction at run time to reduce overhead of large annotation. Weakly supervised methods still require some amount of la-

beled training data and interactive approach require detailed user interaction. We want to eliminate these two disadvantages through our work. We discuss our contribution in the next section.

1.3 Contribution

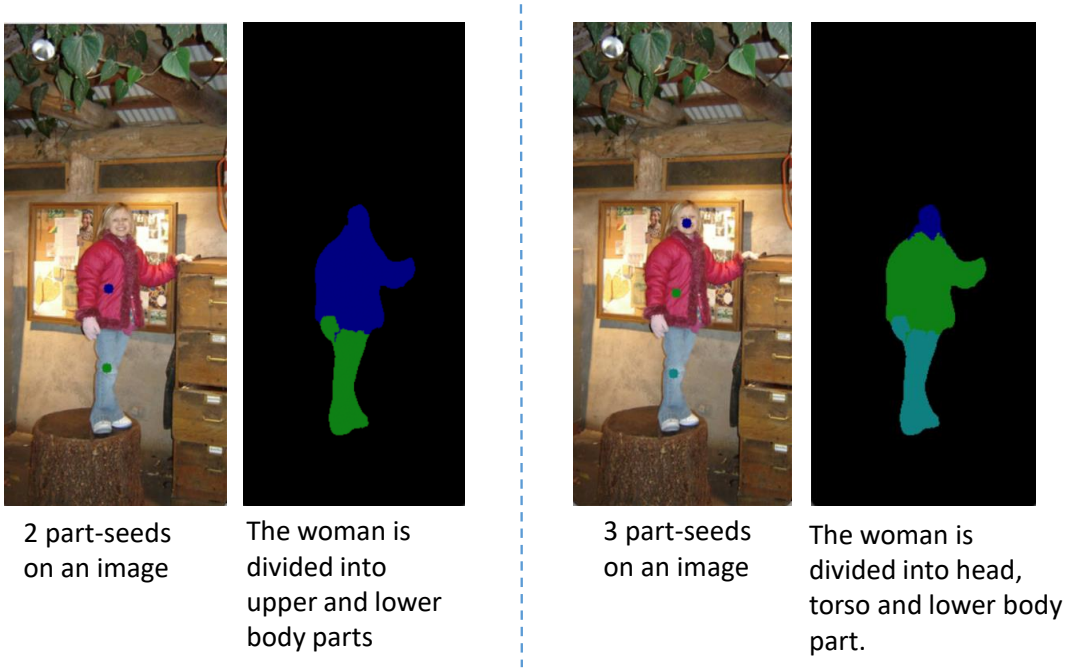

Figure 1.1: Interaction allows the user flexibility to decide how many parts he/she needs and at what level.

While DL-based semantic segmentation networks [31, 24, 19] have the capability to extract parts, they require a lot of training data. Annotating pixels for part segmentation is very tedious, because parts are not well separated by edges. Moreover, part segmentation is a hierarchical problem. As shown in Fig. 1.1, a part (upper body) can be further divided into sub-parts (head and torso), so it makes more sense to have an interactive approach, where user has the flexibility to decide the level at which he needs the parts. Fixing the number of parts and the level of the parts can leave the end-user unsatisfactory. Even in the unsupervised methods [7, 6, 26] where parts are extracted without part-level supervision, although there is flexibility in choosing the number of parts, one doesn't have the flexibility of choosing what those parts should be. Many a time, they don't

provide meaningful parts. At the same time, providing seeds for every image of particular category appears redundant. Thus, interactive co-part segmentation strikes a good balance between the flexibility and meaningfulness one may desire and the manual efforts.

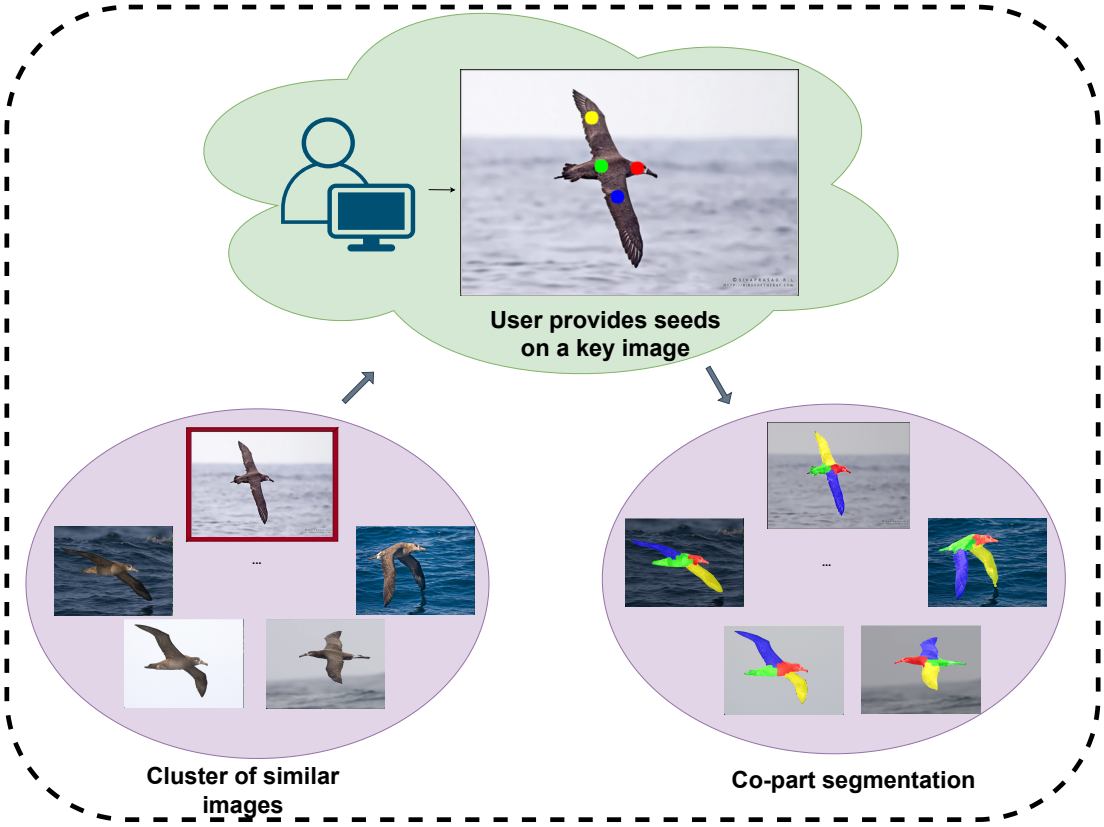

Figure 1.2: Interactive Object Co-part segmentation: Part seeds are provided for a single image in a group of images, but object parts are extracted for all images.

We propose a skeleton-based weakly supervised interactive approach to address these challenges, as illustrated in Figure 1.2. The idea is to have human interaction through seed cues on an image, propagate these seeds to similar images of the same class and label skeleton branches. Once skeleton branches are labelled, part are developed based on matching of branches and with the superpixels. Other interactive co-part segementation techniques [9] have never explored skeletons that provide the object structure, which could be leveraged very well to develop the parts, as demonstrated by our results.

1.4 Organization of Thesis Report

In this chapter, we have introduced the problem, what is lacking in the existing research work, and how do we plan to tackle the problem. In the chapter 2, we will do a deep dive into the existing literature. In chapter 3, the methodology is explained in detail. The ideas that we have tries, rationale behind them and how do they connect with each other. In the chapter 4, we discuss the experimentation and the results. In the final chapter 6, we summarise our work and discuss the future directions that can be taken to evolve it further.

Chapter 2

Literature Survey

Semantic part segmentation is the process of extracting the target object from an image and subsequently dividing the object into multiple parts such that their context represents meaningful information. For example, in the case of an image of the cat, a meaningful part would be extracting pixels belonging to its legs, face, torso, tails, etc. as represented in 2.1

Figure 2.1: A sample image of a horse with its different semantic parts labeled. We can see different semantically meaningful parts such as eyes, tail, torso etc. The image is taken from groundtruth annotation of PASCAL Dataset [1]

Our semantic part segmentation approach consists of many moving parts ranging from image co-segmentation, skeletonization, dense correspondence, and ob-

ject part segmentation. We will go into detail about all of them and how they are useful in our method.

2.1 Image Segmentation and Co-Segmentation

Object Segmentation refers to the process of labelling the pixels of the image such that they are either labelled background or object. Segmentation is really helpful in localising an object in any image and helps machine understand the scenes, the way humans do. There have been multiple classical computer vision techniques in past which have paved the way for image segmentation such as thresholding [32], clustering [33], histogram-based, and edge detection etc. Further algorithms such as graph cut [34], condition random fields(CRF) [35] etc. have been explored.

However it is after the onset of deep learning that we have been able to create architecture which yield highest accuracy on popular benchmark sets. Long *et al.* [36] introduced a fully convolutional network(FCN) which removed the fully connected layers of previous CNN architectures like AlexNet [37] and used deconvolutional layers to predict the segmentation map. This was later imporved by techniques such as U-net [38], Feature Pyramid Network(FPN) [39], PSPNet [40], Mask R-CNN [41], DeepLabv3+ [42], and so on.

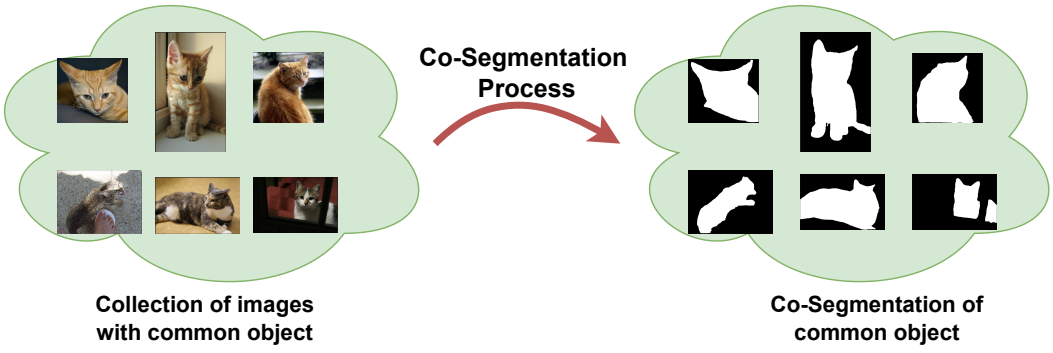

Figure 2.2: The figure shows the process of co-segmentation. The results are obtained using Su *et al.* [2]

In our work, we also utilize one of the pre-trained deep learning architecture for object segmentation because of their low inference time, high accuracy and good generalization over different object classes. Our method has preknowledge

of the class label of the image set which we have to semantically part segment, which helps us explore a niche set of image segmentation techniques known as co-segmentation which are more effective than vanilla deep learning architecture.

The image co-segmentation refers to the task of obtaining a common object across a set of images. We already know our input has a common object across all images which satisfies the condition for object co-segmentation. This can be seen in the figure 2.2. The collection of images have cat as the common object. After the co-segmentation process the object region of the cat is segmented from the background in each image. Recent work in field of co-segmentation includes Deep object co-segmentation [43], DeepCO3 [44], Li *et al.* [45], CycleSegNet [46], Su *et al.* [2], and so on.

2.2 Skeletonization

Skeletonization refers to the compact representation of an object by reducing its dimensionality by one. Skeletonization is a technique that has gained widespread usage in a variety of image processing and computer vision applications, such as shape analysis and recognition, character identification, fingerprint analysis, animation, motion tracking, registration, interpolation, path tracking, medical imaging, and more. In this section we discuss few of the advancements that make the skeltonization stable. An ideal skeleton should be the simplest possible structure of a given shape. The figure 2.3 shows the object mask of an horse and its skeleton extracted using method described in work of Zhang and Suen [3].

Blum [47] introduced medial axis transformation in 1973. The process involves calculating the centers and radii of the largest possible circles that are completely contained within the object and have at least two points touching the boundary. However, the skeleton extraction in this case is deeply troubled by the even slight change in boundary. To overcome this issue, researchers have worked on various approaches ranging from [48, 49, 50, 51, 52, 53]. However these approaches are reliant on manually tuning parameters that controls threshold and other shape

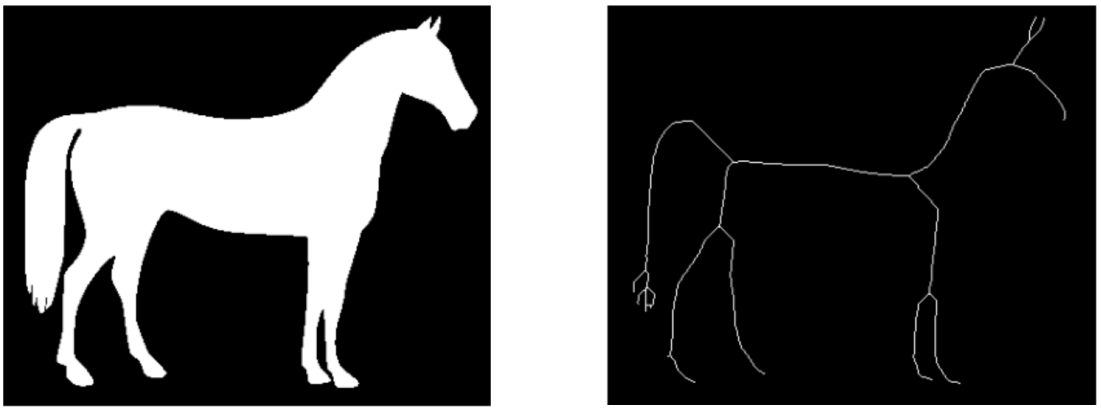

Figure 2.3: The figure shows the object mask of a horse and its skeleton derived using Zhang and Suen [3] method. The figure is taken from scikit-image docs.

related information. This can be a very tedious process and most of the time its really hard to get a clean skeletonization structure for the new shapes.

We can broadly classify the skeltonization algorthims to be one of the three classes-

1. Thinning algorithms - These algorithms are based on thinning the mask of the object in consideration. Arcelli *et al.* [54] proposed a procedure that can efficiently thin an input picture to create a well-shaped skeleton by labeling pixels based on their distance from the background, applying topology preserving removal operations to give it a linear structure, and pruning irrelevant branches. Subsequent improvement were introduced in [55, 56, 57].

2. Algorithm based on Vornoi diagram - Brandt and Algazi [58] approximated the skeleton of a continuous shape based on the Vornoi diagram of the points sampled along shape boundary. The work is improved by [59, 60]

3. Distance transformation algorithms - These algorithms [61, 62, 63, 64]. They propose euclidean distance based mapping for good localization of skeleton points.

4. Mathematical Morphology algorithms - The [65, 66, 67, 68] work on using set of morphological operation to obtain a thinned down version of the

skeleton that contains as much shape and size information as possible. Morphological methods also help in reducing the amount of redundant points in the skeleton representation.

We go into depth for few of the recent siginificant work for stabilizing skeletons. Shen *et al.* [69] introduce a significance measure bending potential ratio(BPR) which is used to remove illegitimate skeleton branches. They improve on the Blum's [47] hypothesis of each skeleton branch corresponding to a contour segment. Shen state that only those contour should produce a skeleton branch which are significant in context of whole contour. BPR gives measure of significance of a contour using both local and global shape information. Tsogkas and Kokkinos [70] focus on contours marking local and approximate reflection symmetry. They extract features representing multiple facets related to each other such as structure, color, shape and spectral clustering information and learn how to combine these cues using supervised learning. The method requires annotated ground truth for a specific application. Shen *et al.* [71] improves on their old methodology of using a bending potential ratio to devise a methodology which stop large structural changes in skeleton on slight contour deformation. Skeleton pruning is used to make the structure simple so that it is unaffected by contour deformation. Their approach differ from other skeleton pruning approaches in the fact that they view skeleton pruning to be a trade-off between skeleton simplicity and shape reconstruction error. Here, author looks for ideal skeleton structure which can have simplest structure while minimising the shape reconstruction error. The algorithm iteratively prunes end branches to increase the simplicity while not letting shape reconstruction error increase drastically.

Latest work in the field focuses on leveraging the power for CNN(Convolutional Neural Network) and deep learning to be able to better represent skeleton. Shen *et al.* [72] propose a fully convolutional network to capture both local and global image context by having multiple scale-associated side outputs. Supervision is used to guide the scale-associated side outputs towards ground truth label. The

CNN based network does not require the object segmentation mask. Zhao *et al.* [73] introduce a hierarchical feature integration technique to tackle skeleton object detection problem. They capture high-level feature from deep layers and low-level features from shallow layers and integrate them hierarchically which helps to refine features and get rich object context and high resolution details. The method requires ground truth annotation for training in an supervised manner. Nguyen [74] introduce modification in the U-Net architecture using attention mechanism which increase reconstruction and feature capturing capabilities. Also, introduces bags of trick for increasing the performance for deep learning based skeletonization model. This method however works on extracting skeleton from the binary images and require training in supervised manner as well.

In the context of our work, we need a skeleton extraction technique which is able to extract a simple skeleton representation from binary segmentation mask These skeleton representation should be rich in object structure information and do not have illegitimate branches. Deep learning and rule-based method that generate skeleton from natural images are prone to have more error in comparison to methods that use binary images. We already extract binary mask using co-segmentation approach which we can use here.

2.3 Dense Correspondence

In our work, we come across scenario where we have to check how similar are same object in two different images and how could we warp one of them to align it to another objects geometry. We would go into deep analysis of deep correspondence networks to achieve the task. The need for dense correspondence network stems from the task of the image alignment. We can see in the figure 2.4 how a source image is warped to align with the target image. Liu *et al.* [75] proposed SIFT flow, a method to align a scene to its nearest neighbour in a image corpus. It involves matching densely sampled, pixel-wise SIFT features between two images. Also, has discontinuity preserving spatial model which allows for matching the objects

which are located at different parts. Based on the SIFT flow, authors design a framework which can transfer image information to a query image according to dense scene correspondence.

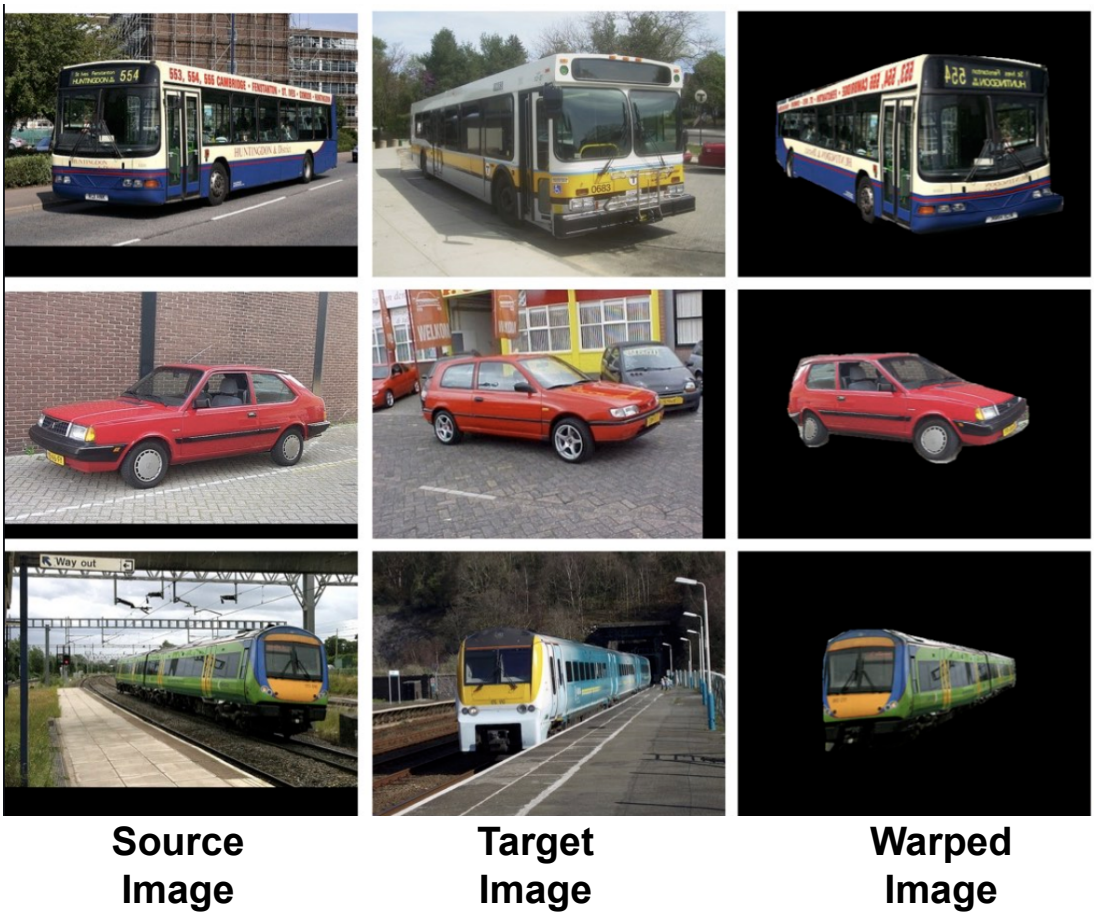

Figure 2.4: The figure shows the original image which is warped to align and match with the target image. The result of warping are displayed under warped image column. The images and warped images are taken from the TSS dataset [4] ground truth.

Here, we would go into detail the dense pixel correspondence between two images which is a task closely related to optical flow estimation task. Optical flow estimation achieves good results for small pixel translation and little appearance variation. We look into the CNN based frameworks that takes advantage of the optical flow methods while also expanding them to instances of large transformations, resulting in accurate estimates with high density and sub-pixel precision. We also check the point correspondence method using pre-trained deep CNN networks.

The traditional way to set correspondence between two images is to detect the keypoints in both of them, compute features of these keypoints and match these feature descriptors. These handcrafted features [76, 77] have limited ability in detecting large variation and changes to the keypoint geometry. There are networks which apply feature descriptor on a pair of images and apply nearest neighbour to match the keypoints globally between them. These approaches [78] do not perform correspondence explicitly and make use of affine transformation or interpolation to convert a sparse set into pixel wise correspondence. However the methods such as [79, 80] are able to match images with large viewing angle difference and large distortion in appearance changes. But they do not perform dense visual correspondence which is our main target here.

Dosovitsky *et al.* [81] proposed a model two construct two CNN networks for estimating optical flow based on the U-Net [38] structure. Ilg *et al.* [82] stacked multiple FlowNet models into a large model which improved the efficieny of the network. This model was termed as FlowNet 2.0. Sun *et al.* [83] proposed a smaller yet efficient model(PWC-Net) than FlowNet 2.0 based on the principles of pyramidal processing, warping and cost calculation. The PWC-Net performs poorly on the very strong geometric transformations. Melekhov *et al.* [84] presented a CNN network for learning dense pixel correspondence between pair of images with strong geometric transformation. Instead of restricting their system to only affine and thin-plate spline (TPS) transformations, as done in previous works, they have trained their model using synthetic data in a holistic manner that can handle a variety of geometric transformations that are typically encountered in the real world. Prune Truong introduced a series of work [10, 85, 86, 87] to increase the accuracy and efficiency of the dense correspondence network by tackling the problems such as pixel accuracy, large displacements, appearance changes, generalization capabilities etc. The issue with deep neural network for dense correspondence task is their inability to generalize well on the unseen class. They are trained for a set task and we need to retrain them either in supervised

or self-supervised manner for another task.

However in our work we are not interested in adding the training overhead of any kind which is why we look at the point correspondence network which have no training phase and use transfer learning to develop an architecture which is able to generalize better. One such work is done by Amir *et al.* [16] where they make use of deep features extracted from the pre-trained Vision transformer [88] for point correspondence task. Vision transformer introduced the method to solve the object classification task using the transformers [89] which were originally developed for the natural Language Processing(NLP) tasks. Vision transformer(ViT) show robustness to occlusion, and less bias compared to the CNN models. Here, we work on the assumption that the features generated for a patch(token) by ViT will be similar to patch feature for the same part of same object in a different image. Amir *et al.* [16] proposed a method to find semantically corresponding points between a pair of images. To reduce ambiguity, the authors suggest using position-aware descriptors obtained from mid-layer features and integrating information from adjacent spatial features through log-binning. They adopt the "Best Buddies Pairs" approach to detect reliable matches between images, keeping only mutual nearest neighbor descriptor pairs.

2.4 Semantic Part Segmentation

Part segmentation methods can be broadly classified into four categories: supervised, unsupervised, weakly supervised, and interactive part segmentation. Supervised approaches make use of labeled training data to train the proposed model. To help model understand the complex data pattern we have to correct it by comparing its prediction with groundtruth. The main issue with supervised approaches is the time it takes to train the models, experimentation required to tune the hyper parameters, and the human workload in intensive ground truth annotation. Unsupervised approaches are trained on only the features and with-

out any explicit guidance or labels from a human. The goal of unsupervised learning is to find patterns, structures, and relationships within the data without any prior knowledge. Unlike traditional supervised learning, which relies on a large amount of precisely labeled data, weak supervision uses sources of noisy labels to train models. For part segmentation, it might have an approach to predict parts with keypoint supervision, object silhouette, etc instead of relying on large amount of quality labeled data. At last, interactive approaches rely on some form of interaction with the end-user for guidance. For example, scribble, point supervision etc. Often, there is an overlap between all the four sub-fields. The methods proposed are mostly a combination of two or three of them. For example, a method might expect the weak supervision in form of scribbles which are produced with the help of user interaction in real time. The weak supervision and unsupervised approaches are mostly combined and used together where the weak supervision is in form of keypoints, bounding box etc. and then we perform unsupervised learning on top of it. We will go into detail for all four categories.

2.4.1 Supervised Part Segmentation

Supervised part segmentation approaches learn the part priors from the labeled training data, which assists part segmentation. Luo *et al.* [23] proposed a Deep Decompositional Network to parse pedestrian images into semantic parts. The author proposed Deep Decompositional Network (DDN) for accurately parsing pedestrian images into semantic regions like hair, head, body, arms, and legs even in heavily occluded scenarios. The approach directly maps low-level visual features to label maps of body parts with good robustness to occlusions and background clutters. DDN uses occlusion estimation layers, completion layers, and decomposition layers to jointly estimate occluded regions and segments body parts. Wang *et al.* [24] proposed approach to tackle the task of object segmentation and part segmentation jointly under the assumption that both tasks are complementary and one information can be useful for the other task. The higher

object-level context guides the process of part segmentation. The method trains a two-channel fully convolutional network which predicts the semantic parts and object potential which is then concatenated and fed to another convolutional layer to predic join object potential. The Semantic part potential is used to derive the semantic parts referring to the semantic parts as the node of the fully connected CRF.

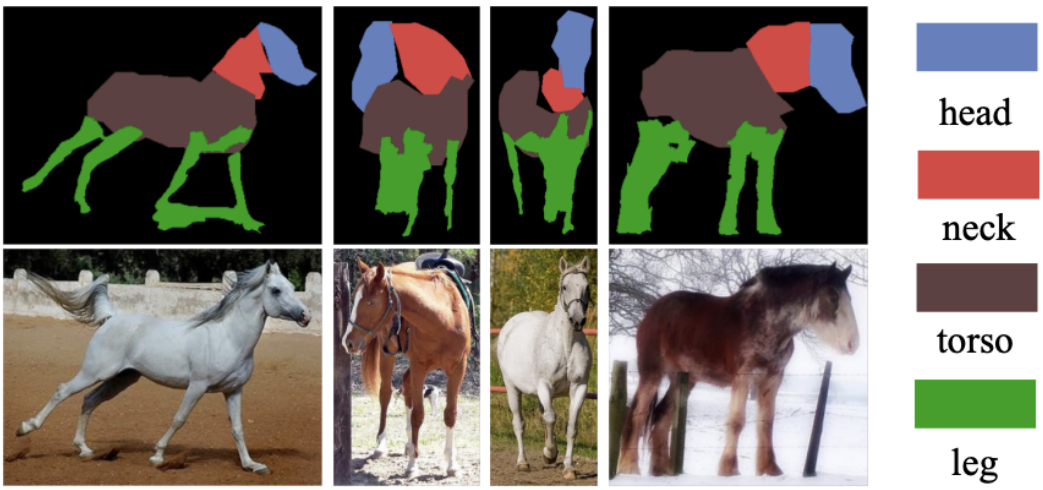

Figure 2.5: The figure shows the results of the Wang and Yuille [5]. The top row shows the learned parts and bottom row shows the original image.

Wang and Yuille [5] proposed an algorithm to learn a mixture of compositional models and these models will be used to represent the boundary of the objects and the parts within it. The compositional models are developed from the edge, appearance, and semantic part cues which are in turned developed using the algorithms such as edge detection and part detection. Author proposes an algorithm that helps in the learning of the compositional models. Lastly, a linear time inference method is proposed to extract the parts from an object. We show the results of [5] in figure 2.5. We can see that the model is able to learn meaningful parts and localize them. Xia *et al.* [19] proposed an architecture to predict the pose estimation and semantic part segmentation of an object. A fully convolutional networks(FCNs) are trained for each task. The two FCNs provide

an initial estimation of pose and semantic part potential, which are fused with a fully-conditional random field to extract refined pose joint location. The refined pose and initial part potential are integrated through another FCN to output semantic part segmentation. Sun *et al.* [25] worked on problem of person retrieval using the approach of part aware identification. They focused on learning part-informed features using the Part-based Convolutional Baseline(PCB) network. The PCB outputs the CNN features which are formed of part-level features internally. The output is processed by another network referred to as Refined Part Pooling(RPP). It corrects within-part inconsistency by recongnising the outliers in each part and reassigning these outliers to a part that they are closest to. Sun *et al.* [25] do not focus on obtaining semantic parts directly however their mechanism gives a glimpse on how these architecture can be employed to directly predict the parts instead of part-aware features.

The supervised algorithm provides good part segmentation but requires sufficient training data, which increases the user workload. Recently, researchers have moved away from the supervised method and are exploring other avenues. This is because the supervised approaches are straightforward and rely on ground-truth to make the model learn, which is always a significant disadvantage. When the scale of input increases exponentially, it becomes impossible to keep up with annotation. To make the methods more generalized and streamlined, where even a new unseen class can be trained and predicted without going through the hassle of creating ground-truth first, new approaches are being explored.

2.4.2 Unsupervised Part Segmentation

Unsupervised methods try to find the complex patterns within data without relying on any labels. Collins *et al.* [17] proposed an idea to localize similar concepts in a set of image. The method finds the details in the features learned from the deep convolutional network and is used to cluster similar features. A pre-trained deep network trained on object cassification is capable of producing features that

can be used as part detectors. Further, Non-negative Matrix Factorization(NMF) is used for data interpretation. NMF is applied on the activation layer, where the output is always non-negative. The result of NMF consists of k factors, which is its pre-defined rank of approximation. We can easily generate the heatmaps for each part given the resutant matrix of NMF application. The method requires for users to input how many parts they want the object to be segmented into. Based on that NMF is applied.

Hung *et al.* [7] proposed a self-supervised deep learning approach SCOPS to tackle the problem of part segmentation in 2019. Most of the work in the field of unsupervised part segmentation is inspired by the work of SCOPS [7]. There are elements of weak supervision in the proposed approach as it works better with object silhouettes as input. Author based the part segmentation problem on four fundamental properties of part segmentation and used them as loss function to make the model earn the intricacies and label the object - 1) A part should be geometrically concentrated. 2) Part segmentation should be robust to any deformation or viewpoint change. 3) A part should be consistent across all the different instances of it, and 4) The union of all parts should form the object. that consists of multiple loss functions based on four principles - geometric consistency, robustness to variations, semantic consistency, and treating the object as the union of parts.

Liu *et al.* [26] proposed incremental work on the idea of SCOPS [7]. The proposed method disentangles the appearance and shape representation of the object and then makes the model learn through reconstruction loss. The proposed method terms a part segmentation as good when it satisfies geometric and semantic constraints which in turn corresponds to appearance and shape of the object. The method improves on SCOPS [7] by removing the dependency on saliency maps to produce good segmentation. The framework takes a pair of image as input - one of them is the transformed version of the other. They both are passed through

different instances of an encoder to extract shape and appearance representation. Their appearance is exchanged and then the latent representation is passed through a decoder to reconstruct the original image. The model is trained mainly on reconstruction loss with other loss supporting the constraints of geometric and semantic consistency.

Choudhury *et al.* [6] also proposed an self-supervised approach on the idea of

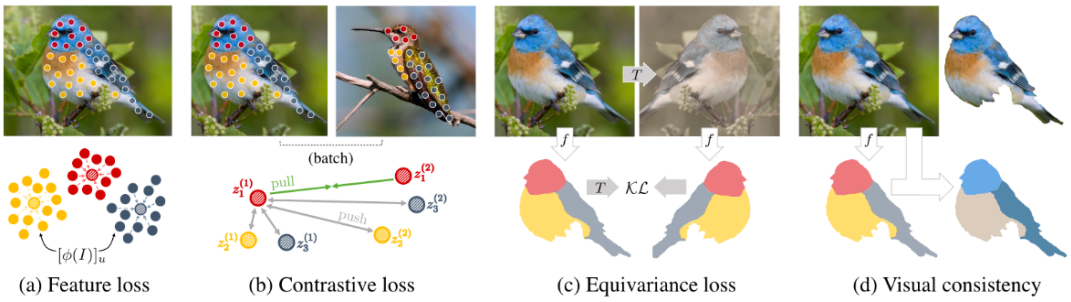

Figure 2.6: The figure shows the different loss function used in unsupervised training as per Choudhury *et al.* [6]. The a) feature loss pulls pixels with similar features together, b) contrastive loss pushes the pixel with different features, c) equivariance loss makes sure that any image transformation does not change the learned parts, and through reconstruction d) visual consistency makes sure the parts make up the whole object.

SCOPS [7]. They introduce new objectives in form of loss function which can help decompose an object into meaningful parts and advocate for the reconstruction loss is useful in this task. They show that keypoint regression, which has been state of the art evaluation metric so far in unsupervised part segmentation, is unable to justify the segmentation quality. Instead new evaluation metric - NMI and ARI are introduced which are able to quantify object decomposition into parts more skillfully. The proposed method applies contrastive constraints on the features of deep CNN layers to formulate parts. As seen in the figure 2.6, feature loss tries to bring pixels with same features together, contrastive loss pushes the different feature locations away, equivariance loss makes sure that a transformed image part labels have same probability distribution as the original image and reconstructs the image to maintain the visual consistency.

The next two work focus on extracting part information from video input. Gao *et*

al. [27] proposes an approach to leverages the motion information of a video feed to extract meaningful parts, the architecture decomposes the frame into parts and then re-assembles it forming a closed loop which supervises the method learning. The parts produced have strong physical meaning in comparison to the previous research efforts in field of unsupervised learning. The proposed approach takes two frame at a time, passes them through an image encoder which outputs the latent feature space, which then is passed to a segment decoder to convert the latent feature into part image. The part image is passed to a decoder to create part mask corresponding to each part, which is then assembled and learning is done through reconstruction loss. Here, as seen in [6] other losses derived from original work of SCOPS [7] are used with the reconstruction loss to create meaningful parts. Siarohin *et al.* [90] also proposed a co-part segmentation approach which leverage object motion in video to extract part information. The architecture consists of two moduls - segmentation and reconstruction module. Segmentation module takes two frames- source and target as input(taken from same video). The segmentation module output the part segmentation map of both image along with affine motion matrix. The reconstruction module reconstructs the target image with the help of source image, segmentation maps of both image, and affine motion matrix. The target image is reconstructed by warping the feature of source image and masking the occluded features using a background mask of target image.

In above approaches [17, 7, 6, 26] , we see that the learned features of a pretrained deep CNN network a really good starting points for extracting meaningful information for part segmentaion. Amir *et al.* [16] takes it one step further and makes use of the features of Vision Transformer(ViT) [88]. ViT has been able to successful surpass the CNN models in recent times and its intermediate deep layers consist of more meaningful information which is shown in the paper. ViT features are clusteres accordingly to segment object in a said number of parts. The proposed approach has no training phase and require only a collection of

Figure 2.7: The figure shows the unsupervised results on the PASCAL VOC dataset [1] as cited in work of Hung *et al.*(SCOPS) [7]. The parts formed are not meaningful in nature.

similar images to perform co-part segmentation. The ability to produce the part segmentation without undergoing a complex training phase(a different model for each number of parts) as in [7, 6, 26] makes the ViT based approach more appealing to researcher who want to use the method as it does not add any overhead.

Unsupervised method fail to create meaningful parts which can be seen in the figure 2.7. If we compare the result of class sheep, the torso is divided into two parts which do not hold any meaning. The parts are divided based on consistent regions across objects of same class. It does not take into account if the part is meaningful in itself. Also, the number of parts are predefined which forces the object to always be segmented into said number of parts. The biggest drawback of the unsupervised method lies in its failure to recognize meaningful parts.

2.4.3 Weakly Supervised Part Segmentation

The weakly-supervised learning methods require weak supervision in the form of weak labels such as keypoint, bounding boxes, image-level labels, eye tracks, heterogeneous annotation and so on. The idea of using weak supervision for part segmentation derives from its usage and success in semantic segmentation [91, 92, 93, 94, 95]. The main purpose of weak supervision is to balance

the trade-off between test time accuracy and training time annotation cost of ground truth label. Pathak *et al.* [91] trained a fully convolutional network with weak image label supervision. The author formulated a multi-class logistic loss function for the training. Dai *et al.* [93] proposed an approach which is able to output competitive results in comparison to supervised methods with the weak supervision of the bounding boxes. The model generates multiple region proposal for each bounding box and the convolutional network is trained for each region proposal. Bearman *et al.* [94] proposed an approach where annotator point(one or few pixels) to whether an object exists in the image and where it is. Papandreou *et al.* [95] proposed an approach that uses a combination image-level labels along with bounding box to train the deep neural network. The weak supervision combined with few strong supervised data points help in guiding semantic segmentation.

The weak supervision based part segmentation follows the same principles as the weak supervision based semantic segmentation and makes use of keypoints, image-level labels and bounding box to help localise the object and segment the parts. Yang *et al.* [28] developed an approach to parse set of clothing images to meaningful different cloths. The author proposed an approach to extract semantic region from the image using weak image-level supervision through technique of co-segmentation. A graphical model is constructed with the semantic image regions as vertices and through Graph Cut algorithm, the task of joint part label assignment is performed.

Krause *et al.* [15] proposed an approach for the fine-grained recognition task based on part annotation. The part annotation here is generated with the help of co-segmentation and alignment combined in discriminative mixture. The assumption here is that object of same category share similar shape which makes alignment simler and can be done with the help of just object segmentation. Object co-segmentation is performed using GrabCut [96] technique with the help of image-level label and bounding box. The images to be aligned are decided

with the help of finding cosine similarity between the image feature from fourth convolutinal layer of the objects(features are robust to changes). The similar images are aligned via a dense alignment function. The parts are generated through aligned images based on how diverse they are. The generated parts are then used for fine-grained recognition.

Meng *et al.* [11] proposed an approach to extract local part segmentation with the help of assumption that each local part will remain fixed along the object pose variation. Weak supervision is incorporated via object tag and pose tag of a set of images. For example a set of images labeled 'Cat' for object tag and 'Standing' for pose tag is used. The method considers similarity for object co-segmentation, shape matching for part detection and then graph matching for part assignment. The authors define a energy function for part-labelling the object pixels via four terms- the image segmentation term and region consistency term defines what is foreground in the image, the part consistency terms define what is a single part in all the images through the shape variation and global part structure consistency term defines the similarity between part structures throughout the images. The optimal solution is achieved via the energy minimization of the formulated energy problem. The final result is decided by number of parts to segment the object into. The work depends greatly on the assumption that objects have similar features and when they differ, the foreground consistency term is unable to define the object silhouette correctly.

Fang *et al.* [8] proposed an approach for the human body parsing via weak supervision in form of keypoint annotations. The method looks for human objects with similar pose and transfer the parsing result of a human object to another one with similar pose. The assumption here is that the human objects with similar pose should have similar part segmentation as well. For each test image as seen in the figure 2.8, we find the images in training set with similar pose. For each selected training image, pose guided morphing of part segmentation(labeled

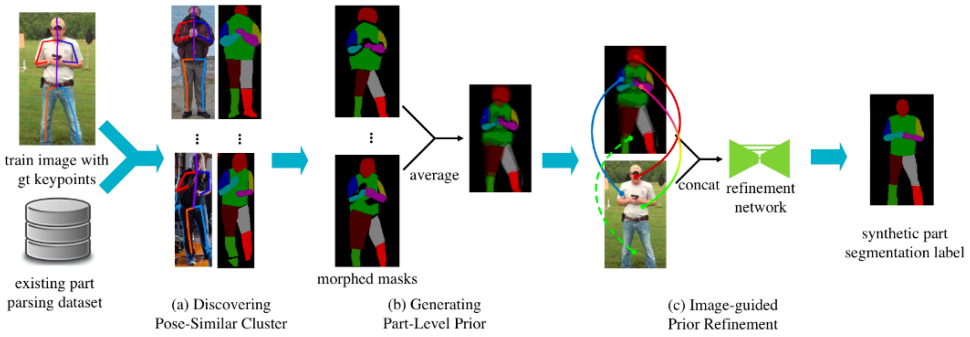

Figure 2.8: The figure shows the framework of Fang *et al.* [8]. For an image with keypoint annotation, similar images are retrieved from database. The part level prior is obtained by applying morphing on part masks of all the retrieved images. An image guided refinement network learns to produce final part segmentation with the help of morphed part prior.

ground truth) is performed to align it with the target pose as shown in part b) of the figure 2.8. The aligned part segmentation proposal are average to form part-level prior for target image. In the figure 2.8 part c), the averaged part-level prior is passed through a refinement network to estimate the final part segmentation. The refinement network is a 'U-Net' [38] like structure trained on L1 loss between proposed binary mask of each part with ground truth annotation of the parts. The network is trained only for the training images which have both the pose and part annotation. These images will be less in number. The advantage of having a strong part-level prior is that the refinement network can be trained with few part annotated training example. Weak supervision in this method is in form of keypoint annotation of each image(training and testing), and part annotation of few training images(the number of annotated images are very less compared to the supervised methods). We can see the results and weak supervision used in the Fang *et al.* [8] in figure 2.9. The morphed part prior mask is the culmination of all the part labels of the objects with similar keypoints pose. The morphed part mask is used to train a refinement network which results in part segmentation results shown in third column. Here, we can see how weak supervision in form of part prior help segment the object.

Naha *et al.* [29] proposed an approach which builds on the work of Fang *et al.* [8]

Figure 2.9: The figure shows original image, weak supervision, and the final part segmentation results. The weak supervision is in form of morphed part priors. The image is taken from the work of Fang *et al.* [8] and shows their results.

and improves its shortcomings. The issue with [8] is that its approach to kepoint based part creation fails in case of complex poses and is unable to generalize across different object classes such as animals. The author removes the need to find similar pose objects and averaging their part annotation prior. Here, the keypoint annotation of target image is directly used to produce pseudo-part segmentation which is then refined with help of appearance information of the object to output improved part segmentation. The network is divided into two parts- the Pose to Part CNN network used for producing the pseudo part segmentation and the Visual Evidence CNN network to refine the pseduo part segmentation. The Pose to Part is a U-Net [38] with encoder-decoder along with skip connections. The idea is similar to [8], that the se of keypoint annotation reduces the number of training images required to train a efficient network. The method proposed is tested on the quadruple classes of the PASCAL Part dataset [1].

2.4.4 Interactive part segmentation

Interactive part segmentation refer to approaches which include user interaction. Interaction can be in form of scribbling, point supervision/clicks/cues, etc. Similar to weak supervision approaches, the interactive segmentation is widely used in the semantic segmentation of the object and the idea of using it for part segmentation derives from the same. Early roots of interactive segmentation can be found in the work of Boykov and Jolly [97], who used the interaction in form of marking certain pixels defining what is foreground and background. Based on the hard constraints provided by these pixels, a graphical model is formed which is solved by Graph Cut algorithms to find an optimal solution. Batra *et al.* [98] proposed a co-segmentation approach to segment the common foreground object from the set of images using the interaction in form of scribbles. The user decides what foreground object is and scribbles on certain parts of it which defines the segmentation.An energy function is formed using the different terms based on

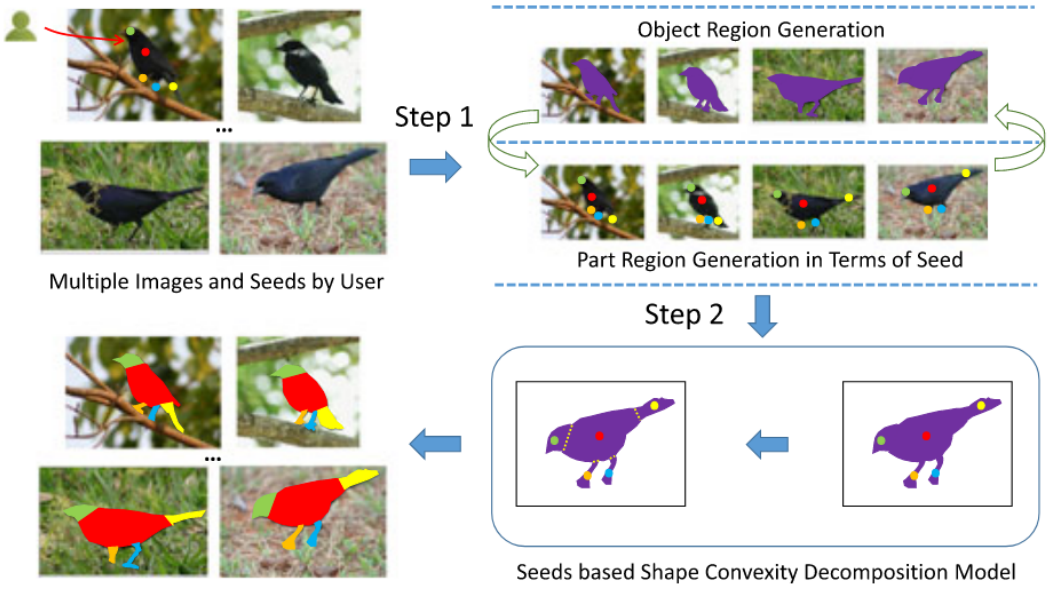

Figure 2.10: The flowchart of the approach proposed by the Meng *et al.* [9]. The image is taken from the work of Meng *et al.* [9]. User seeds are propagated to rest of the examples. Shape convexity decomposition is performed to generate parts.

scribbles and a graph cut solution is used to obtain the optimal segmentation output. It is found that the method works better when scribbles are in the hard to detect areas. Further down the line, Lin *et al.* [99] proposed a scribble based convolutional approach for the semantic segmentation. For interactive part segmentation, there is a little literature to discuss, We discuss two of such works which incorporate the interaction to produce semantic part segmentation.

We study in detail one of the most prominent paper on the interactive part segmentation titled "Seeds-Based Part Segmentation by Seeds Propagation and Region Convexity Decomposition" by Fanman Meng [9]. Meng *et al.* [9] proposed an approach that incorporates user interaction in the form of seed cues which further assist in semantic part segmentation via shape decomposition. The shape convexity analysis driven by seed cues provides decomposition and part segmentation. We will go in details how the method is formulated, what are its novel ideas, and what are the drawbacks. As seen in the figure 2.10, the input to the algorithm is a set of images with one image with seed cues(i.e. a pixel

for each part of the common object). Two main aspects of the method are - 1) How are seeds propagated to the other images in the set, and 2) How parts are generated from the seeds on the object of an image. For the first task, the shape similarity of the the image with seed and the target image is studied with consistency constraints for each seed. For second task, author has combined the seed location with the other information such as common object region among all the images and convexity analysis of the part regions. To summarise, we have a group of images with common object as foreground. One image is chosen for user interaction, where user gives seeds(a single pixel referring to the location where a part is located). User gives seeds equal to the number of parts they seem the object should be divided into(the final result depends on the users discretion). Now the seeds on the source image are propagated to all the other images via a seed propagation method. After that, part generation is done for common object within each image.

The generation of part seeds and what should be the common object region is done simultaneously complementing each other. The object is defined by three constraints - seed matching term(it evaluates the consistency of seeds linked by dense pixel prediction), a term to enforce the seed to be within the object region, and finally a term for to check similarity between object region of the images. All three of them are evaluated together. Seed and object region corresponding to the minimum value of the function are chosen as optimal seed location and object region(i.e. object silhouette). After the above process, shape decomposition of the object region is performed for part segmentation. The shape of the object region is decomposed into several convex local regions based on cutline selection as it can be seen in the Step 2 of the figure 2.10. Author adds seeds constraint to the shape decomposition objective. The constraints are - 1) each seeds needs to be in a different part, 2) each region cut should have small cost, and 3) each part should keep convexity.

The drawbacks of the proposed method are reliance on object region for part gen-

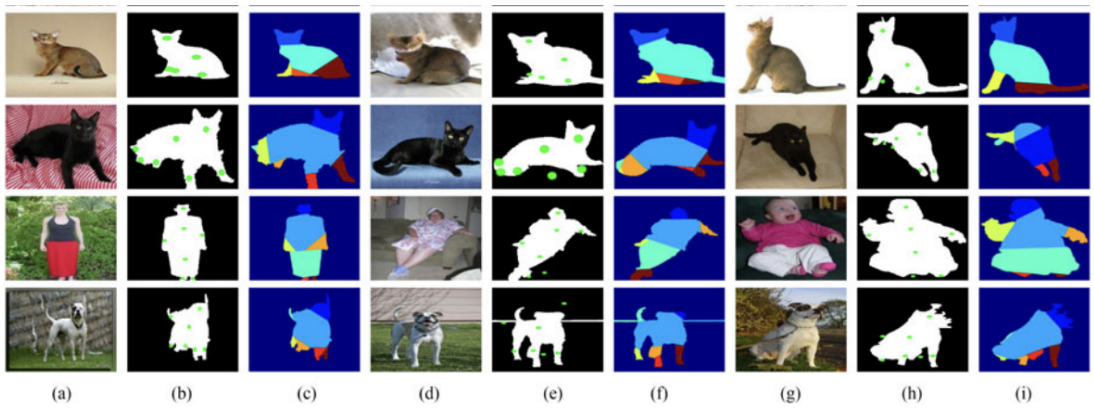

Figure 2.11: Part segmentation results by the proposed method. (a), (d), and (g): the original images. (b), (e), and (h): the seeds. (c), (f), and (i): the part segmentation results. Note that (a) and (b) display the initial images and the seeds drawn by user. The image is taken from the work of [9].

eration. If the object region is not segmented properly, it overthrows the whole algorithm and the final part generation are not proper. The boundary of the parts do not look natural because the method is based on the cutline selection. Seed propagation depends heavily on the object region as well. The methods for object region generation are not efficient as the overall jaccard similarity of the datasets evaluated in paper are 0.742. We can see in figure 2.11 that the part segmentation results do not have natural boundaries and are not able to define a part well. Few of the seeds are propagated out of the object region. These are the shortcomings we address in our work.

We also discuss a Generative Adversarial Networks(GANs) [100] approach used for weak supervision proposed by Tritrong *et al.* [30]. Tritrong *et al.* [30] explores the question whether GANs are able to learn meaningful parts while trying to construct image. The architecture is divides into two parts - a pre-trained GAN is used to extract feature for the test image and these features are then passed through a segmentation network to produce the part segmentation result. During the training phase, we leverage a pre-trained GAN for that specific class to generate training examples. Few of these examples are presented to the user and user annotated these examples with the part information. The user-

interaction decides how many parts are to be generated and how do they look. These part annotations are used as ground truth for the segmentation network which is CNN or MLP network. During the test time, each image is passed through the generator and features are sampled at intermediate layers. The sampled features are then passed through segmentation network to produce part results. This approach requires training a GAN for each class separately which a very computationally costly task. The test image distribution should not differ much from the training distribution else the features generated are not much useful. The testing phase takes time which increases the time complexity of the model.

As mentioned above, there is not much work done in field of interactive part segmentation and most of the inspiration comes from the work of interactive image segmentation. We take inspiration from the work of Meng *et al.* [9] on what should be the form of interaction, outline of the tasks, a closer look at the shortcomings and what areas to deep dive in. We propose an interactive weakly supervised algorithm that uses objects' skeleton masks(weak supervision) as the basis of part segmentation, and the seed cues shared by user(interactive supervision) drive the labeling of skeleton branches.

Chapter 3

Proposed Methodology

We provide a detailed discussion of our proposed method in this section. We start with an overview and then discuss different components involved.

3.1 Method Overview

We aim to segment the shared objects in an image group into meaningful parts with the assistance of seeds cues on one of the images via skeletonization approach. As shown in the flowchart shown in figure 3.1, the proposed method accepts a group of similar images as input. It solicits seed cues(referring to pixel location) from the user for each visible part of the object on the key image. The key image seeds are propagated to the remaining images using a correspondence method. Object segmentation and skeleton masks are extracted using co-segmentation and skeletonization, respectively. Branch labeling is implemented using the updated current image seeds and skeleton. The co-part segmentation is generated through object region propagation.

We will discuss how to generate the required inputs for the algorithm such as co-segmentation mask of the images, skeleton outline of the images, and propagated seed cues for each image Once we have these information, we will move onto the branch segmentation and labelling process to generate the effective part segmentation. At the end, we discuss the implementation details of the network and how to deal with the large image set.

Before deep-diving into a detailed explanation of each subpart, we would go through the notations used. The below notation is for each cluster. A set of n similar images belonging to the same class is denoted by $I = \{I_1, I_2, ..., I_n\}$. The

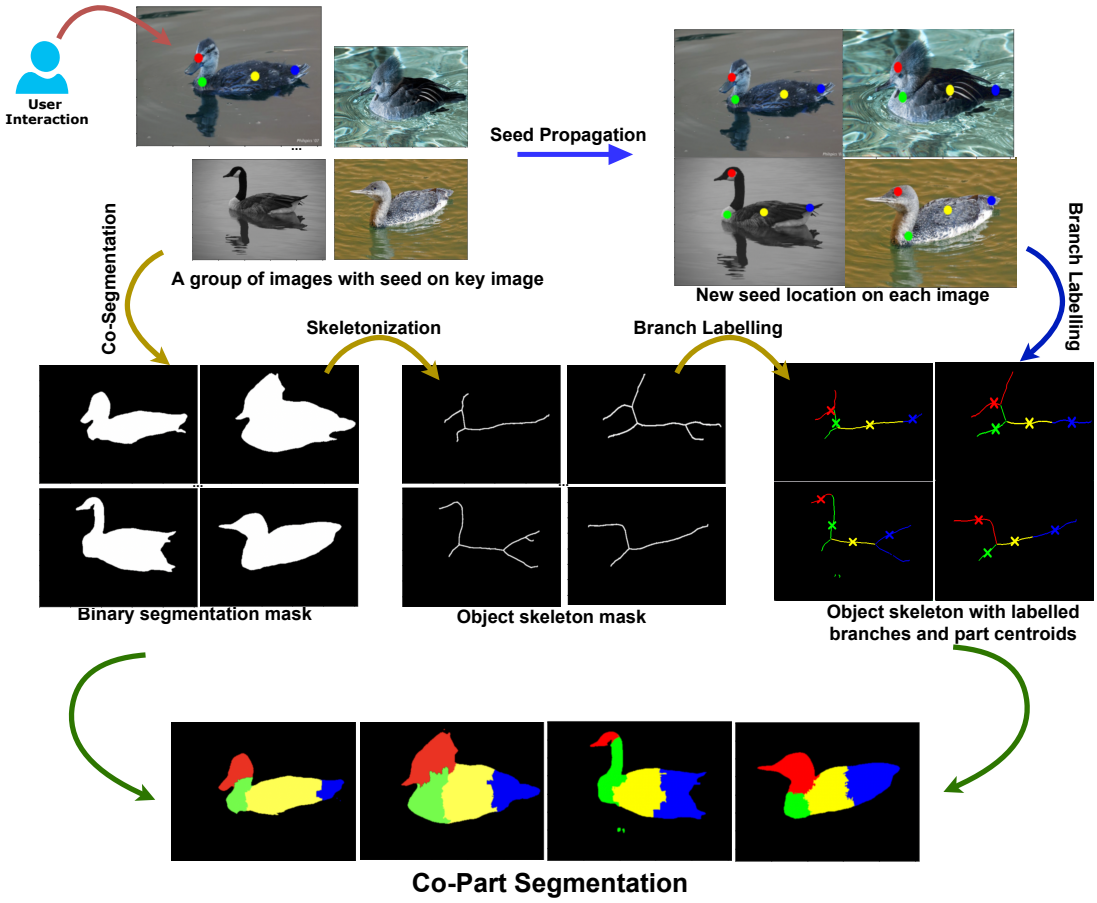

Figure 3.1: Flowchart of proposed approach. Input: A group of similar images, Step 1: User interaction with key image , Step 2: Co-segmentation and co-skeletonization, Step 3: Seed propagation, Step 4: Labelling of skeleton branches using seed cues and unsupervised mask, and Step 5: Co-Part Segmentation

co-segmentation mask and skeleton mask are denoted by $O = \{O_1, O_2, ..., O_n\}$ and $K = \{K_1, K_2, ..., K_n\}$ respectively, where O_i and K_i are the co-segmentation mask and skeleton mask of the i-th image. The seed cues provided by the user are denoted by $S_1 = \{s_{11}, s_{12}, ..., s_{1m}\}$, and m is the number of parts in the object as specified by user. The S_i represents the seed cue for the i-th image and is denoted $S_i = \{s_{i1}, s_{i2}, ..., s_{im}\}$. Our output would be $P = \{P_1, P_2, ..., P_n\}$, where P_i refers to the part segmentation output of the i-th image. Each output would be divided into m parts where m would depend on the number of seeds provided on the key image and it is denoted by $P_i = \{p_{i1}, p_{i2}, ..., p_{im}\}$.

3.2 Object co-segmentation and skeletonization

We already have image-level label for the set of images. It is already known that the images have a common foreground object which we need to segment. In such cases the co-segmentation approach can be more useful than the traditional object segmentation approaches. Co-segmentation approaches can leverage the pre-known knowledge of having a common object to segment and use the feature similarity of the object to a great extent. One such method is proposed by Su *et al.* [2]. that uses transformers to view the image feature as a patch token, and captures long-range dependencies through the self-attention mechanism.

The skeleton mask of an object is the basis of the semantic part segmentation method. We look for skeletonization approaches based on object mask as input. As seen with the object segmentation, the knowledge of image level labels helps in the task. In the same way, we look for approaches that can produce object skeleton through object silhouette. There is no additional overhead in the task as we have already extracted the object silhouette through the co-segmentation. Using the deep learning approaches that work directly on the original image [73] is not advisable. The result of such approaches is poor compared to silhouette based approaches because the techniques that work on original image, go through extra steps of detecting the object, examining its boundary, extracting the object region and then works towards extracting the skeleton. However the object silhouette based approach already have the base object shape to work with and it improves on the boundary of the object. In our work, we have extracted the skeleton through Shen *et al.* [71]. [71] proposed a method that deals with the tradeoff between skeleton simplicity and shape reconstruction error and it tackles the issue of large distortion in the skeleton shape with slight contour deformation.

We extract segmentation masks O from the set of Images I and skeleton masks K from the set of segmentation masks O. Co-segmentation masks O and skeleton masks K would act as input for our algorithm along with image set I.The segmentation and skeleton mask output can be seen in the figure 3.1.

Figure 3.2: Few key images with seed cues obtained from user interaction. Each seed cue corresponds to a distinct part.

3.3 Seed Generation

The seed generation task is not complicated and dependent on how the user perceives the object and how detailed they want to be with the request. Here, we also introduce the idea of the key image. Key image is the one user interacts with and it is in a way representative of the image set as whole. The user marks the part location on the key image.

For simplicity we take example of a set of images of horse. The user interacts with the key image of the set. An image of a horse might be such that it has a head, neck, and torso, but the legs are not visible. Another image might have legs, but the tail is missing. We assume that user will correctly identify the parts present in the image they are interacting with, and for each such such part, they will provide a seed cue. A seed cue is a click or a single coordinate point of a location residing somewhere in that part. A seed cue set of key image S_1 will have m seeds in it. The m is a variable number that depends upon the image on which seeds are taken. If an image has m seeds, it signifies that the user identified m semantic parts in the image and added a cue for each part.

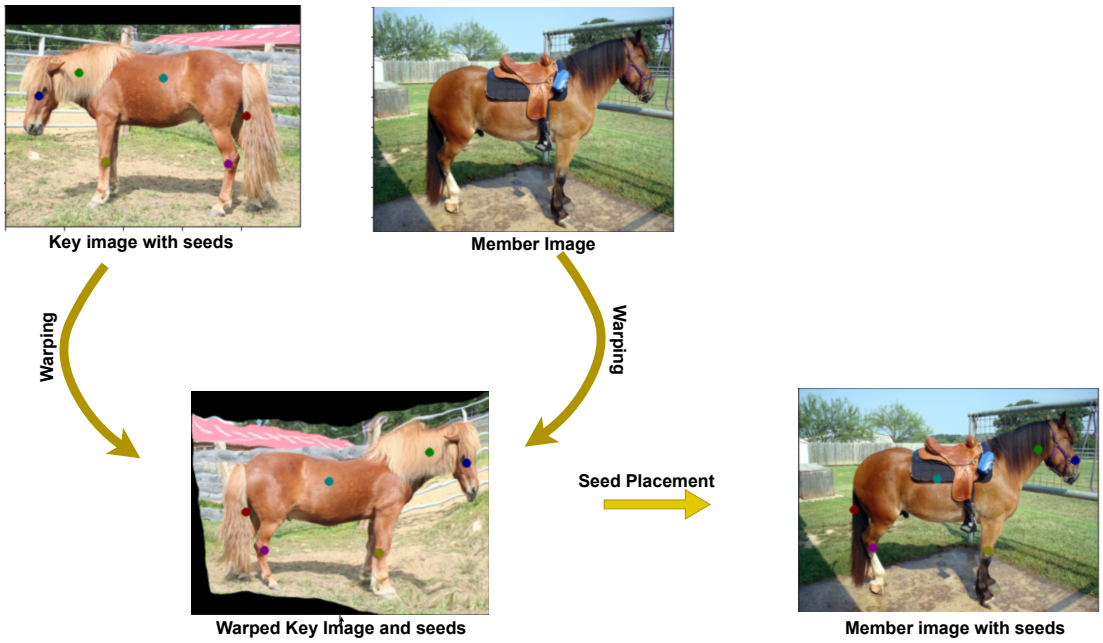

Figure 3.3: The above figure shows how the source image is warped in the target image and new seed locations are retrieved using GluNet [10]

We can see few examples of key image and the parts marked by the user in figure 3.2. The colored dots refer to the seed location i.e. a single coordinate location within the part. In figure 3.2, for the image of the baby, the blue dot refers to the head part location, the green dot refers to the torso part location, the red dot refers to the left hand part location, and so on.

3.4 Seed Propagation

The user provides a seed cue on only the key image. We perform seed propagation to obtain seeds on rest of images with reference to key image seeds.

3.4.1 Seed Propagation: Dense Correspondence Network

First we worked with deep learning based approach suggested by Truong *et al.* [10]. We perform correspondence between the cluster representative and other images in the i-th cluster. The objective is to warp the cluster representative R_i to the target image C_{it} on which we want to obtain the seed. Once we have a

warped image W_{it}, we calculate the displacement of all seed coordinates, and the new location of those coordinates after the displacement act as the seed S'_{it} for the target image. For the i-th cluster-

$$W_t,\ disp_t^{map} = \omega_{s->t}(R, C_t)\ \ \forall_{t \in 1...k} \tag{3.1}$$

$$s_j^* \ = \ \arg\min_{s^*}\ d(s_j, disp^{map})\ \ \forall_{j \in 1...m} \tag{3.2}$$

In equation (3.1), $\omega_{s->t}$ is warping function that warps the source image to target and outputs the warped image and the displacement map. Its input are the key image(R) which is transformed and each of k member images(C_t). In equation (3.2), we calculate the new seed locations of a single image using its original j-th seed location(s_j) and its displacement map($disp^{map}$). In equation (3.2), d is a function that calculates the distance of each point in displacement map with j-th seed location and returns a distance probability map. The location with minimum value is taken to be the new seed position for target image. The distance probability map is smoothened with gaussian blur before finding argmin to cover up false peaks. This way new location of all m seeds is deduced.

We can see in figure 3.3 how seed are propagated from the source image to all the other images in cluster via a sample example. The issue with the GLUNet [10] is that the images with complicated and complex orientation are not properly warped which causes the seed propagation to have high error. The orientation is not captured well if its in different direction. This leads us to explore other approaches.

3.4.2 Seed Propagation: Point Correspondence

We explored the latest work to see what could better fit our problem and generalize well even for complex pose and orientations. We use the correspondence method described in [16]. The idea is to use a vision transformer(ViT) [88] and find the features of patches in both images. Each patch is matched with the

most similar patch in the target image via cosine similarity. Once we have the patch mapping, we find the patch number in which our seed is lying, and subsequently the most similar patch for it in the target image. The center of the target patch is treated as the new seed location. We, do this for all the seed cues in original image. We also use the co-segmentation mask to guide the seed propagation. The patches that fall under the co-segmentation mask are termed as foreground patches and similarity is computed only for those. This constrains the seed matching to look for new seeds in the object region and not mistake background patches as the new location.

$$(p\_x, p\_y) = 1 + ((x, y) - padding)\,//stride \tag{3.3}$$

$$(tgt\_p\_x, tgt\_p\_y) = similarity\_map\,(p\_x, p\_y) \tag{3.4}$$

$$(new\_x, new\_y) = (int\,(tgt\_p\_x, tgt\_p\_y) - 1) * stride + stride + \frac{padding}{2} \tag{3.5}$$

The method is run for each seed cue for a pair of images(source with seed cue and target image). For a seed cue, we calculate the patch location in source image by the equation (3.3). (p_x, p_y) refers to the patch location in a 2-D grid of source image patches. For the source patch, we find its most similar patch in target image using the cosine similarity grid generated for both the images. The most similar patch in target image (tgt_p_x, tgt_p_y) is termed as the patch where the target seed cue should be. The similarity is calculated on the features of each patch, the feature of the patches is similar when they are representing similar information of the common object in two images. Using the equation (3.5), we calculate the exact location of the new coordinated of the seed cue for the target image.

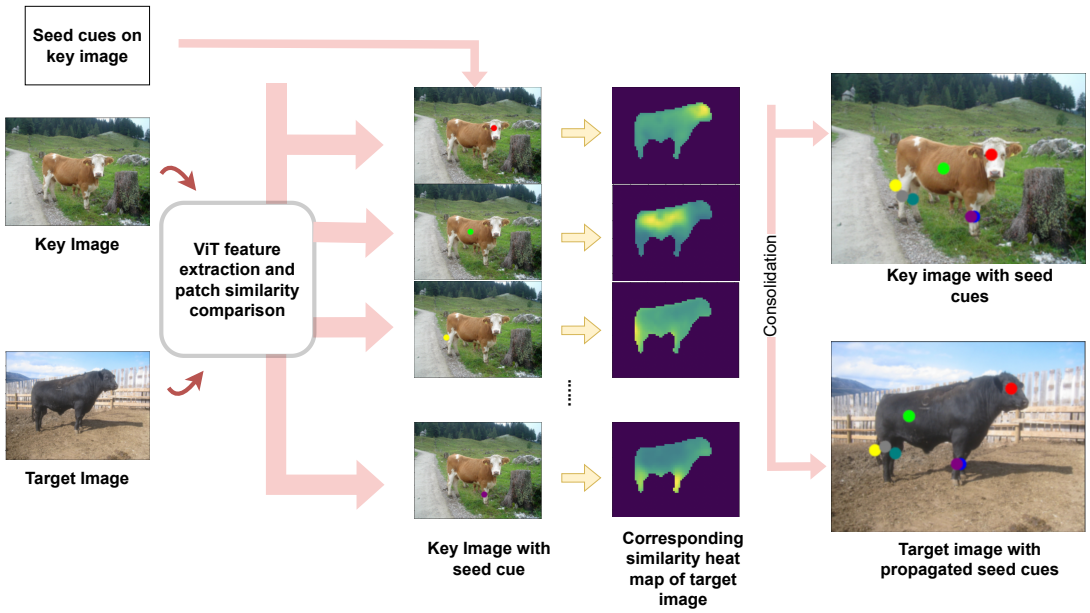

Figure 3.4: Different stages of seed propagation: Stage a) Source image with user's seed cue and target image, Stage b) Feature extraction and patch similarity detection , Stage c) Generating heatmap for most similar patch corresponding to a seed cue , and Stage d) Consolidating all the seed cues

The proposed approach can be seen in the figure 3.4. First the key image and target image are passed through ViT to get the patch configuration and their features. Now for each see on key image. we generate a similarity heat map to show what area is most similar in the target image. As we can see for the seed on the leg, similarity heat map shows that the all the legs in target image are generating high intensity as compared to other parts of the object which shows that the algorithm works as expected. Finally, all the new seed cue location on target image are consolidated.

3.5 Skeleton Based Part Segmentation

At this stage we have the object co-segmentation, object skeleton, and the seed cue for each image. As seen in the figure 3.1, we use these details to move onto the next process which is branch labelling and co-part generation. In this section, we study the process of part segmentation in detail. We will refer to the figure 3.5 throughout the section for discussing the our approach.

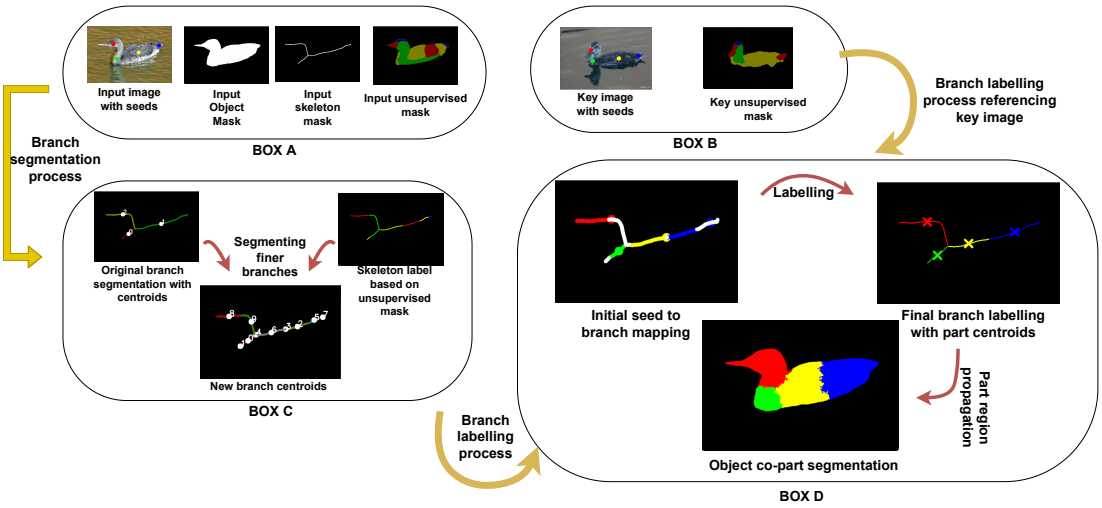

Figure 3.5: Different stages of part segmentation. Box A) Input image image with seed cues, co-segmentation, co-skeleton, and unsupervised mask, Box B) Key image seeds and unsupervised mask for refrencing during labelling process, Box C) Finer branch segmentation , Box D) Branch labelling and part region propagation to generate object co-part mask

3.5.1 Skeleton Decomposition

Skeleton decomposition proposes to exploit the skeleton's topology to quantify the object into semantic parts. However, using the skeleton mask directly for seed labeling is not practical. As illustrated in Figure 3.5's branch segmentation process, we segment the skeleton into branches. Skeleton decomposition produces branches that correspond to specific parts of the object. The decomposition relies on identifying the corner points where multiple branches meet to segment the skeleton into distinct branches. Box C of Figure 3.5 shows the initial branch segmentation. In the example, there is only a single joint which leads to skeleton decomposition into three branches.

Some skeleton branches can be long and encompass multiple parts of the object. If an extensive branch is assigned with one seed label only, it can negatively impact the labeling process. For example, in a specific scenario, the skeleton branch of the leg can encompass parts of the torso which can result in the torso region getting classified as the leg. To evade such a scenario, we take the help of unsupervised mask prediction. We check how many unsupervised mask parts lie on that skeleton branch and use it to decompose the skeleton branches into finer

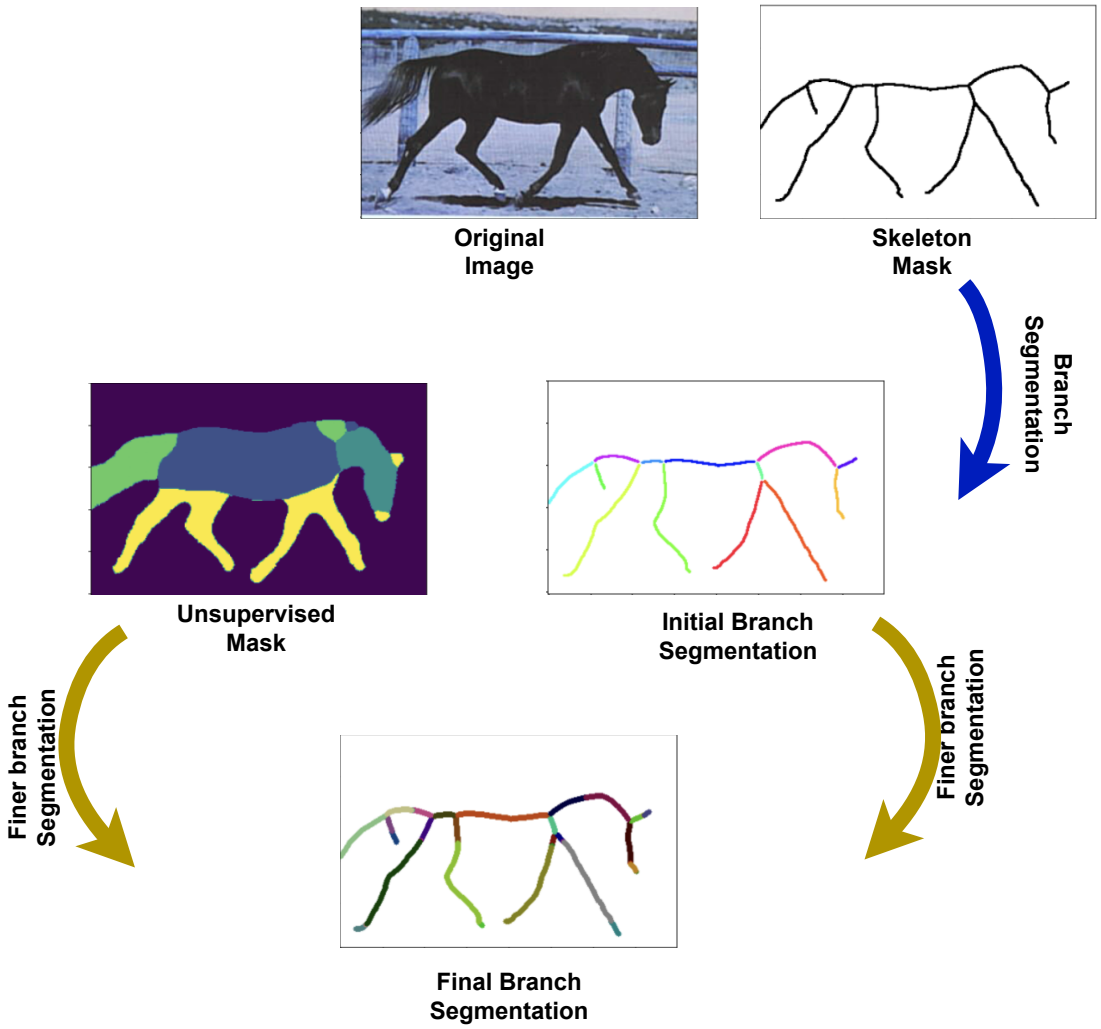

Figure 3.6: Different stages of branch segmentation. The skeleton of the object is used to generate branches. The branches are then segmented into finer versions with the help of unsupervised mask.

cuts.

This process of segmenting into finer branches is illustrated in Box C of Figure 3.5. Working with smaller branches ensures that incorrectly classifying a branch will have a minor impact on the overall part-labeling process.

We have, so far, discussed how we take the skeleton of an object and divide it into different branches and then further refine those branches with the help of an unsupervised mask. We will take an example here to see the process up close and in detail. In the figure 3.6, the skeleton of the horse is segmented into different branches as seen in the 'Initial Branch Segmentation'. Each branch is colored distinctly. Lets closely examine the initial branch segmentation. The

head and the neck region are in different branches which would make it easy for us to label them in separate parts. The same goes for the tail portion. However the legs branches are covering a large region of the torso. So if the leg branches are directly labeled separately, they will encompass a large portion of the torso. This is one of the scenario where we need the finer branch segmentation. Now to segment the branches further, we take the help of unsupervised mask. The unsupervised mask is for 4 parts. So, it creates four region in the object. We take help of this labelling. In the unsupervised mask, all the legs are of same label and they do not interfere with torso region. We segment each initial branch further using the different unsupervised label in that branch. The 'Final Branch Segmentation' shows the finer segmentation of branches. The leg region is now further divided into smaller branches which can be labelled separately.

3.5.2 Skeleton's Branch Labeling

We aim to label all the skeleton branches with a unique seed corresponding to a specific object part. While the straightforward approach is to match the seed coordinates with branch centroids or pixels, this method may miss some seeds or assign incorrect labels due to misplaced seeds during propagation.

We designed a more sophisticated approach to maintain object part consistency throughout the image set. We use the key image as a reference to match skeleton branches with their corresponding seeds in the current image. This involves utilizing both images' unsupervised masks in the labeling process. The unsupervised masks help determine the set of branches and seeds to compare. The unsupervised mask ensures consistency across images. For example, if we have a mask with four parts, as shown in figure 3.5, a specific unsupervised part aims to reference a similar region in all the images, thereby reducing errors in estimating new seed locations via correspondence.

For each seed, we identify its label in the unsupervised mask of the key image and then restrict it to only compare with branches that have the same label in

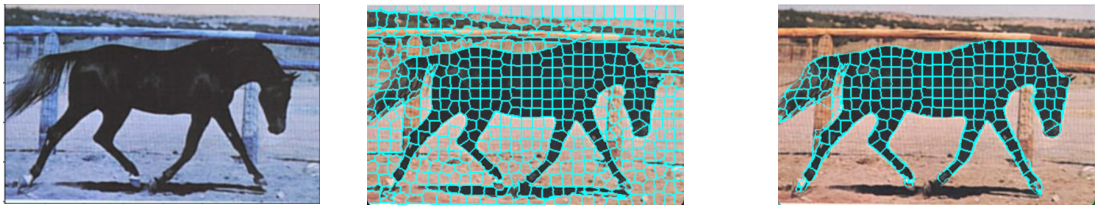

Figure 3.7: Different stages of superpixel generation. The SLIC superpixels generated from original image. Further, only the superpixels in object region are marked for generating parts. We discard the superpixel that belong to background.

the unsupervised mask of the current image. This results in branch and seed sets for each unique label of the unsupervised mask to compare. Using the euclidean distance of seeds with branch centroids, we map the closest pair to get the initial seed-branch mapping, as shown in figure 3.5 box D.

For leftover branches, we check their associated branches. If only one of its branches is labeled, we assign its label to the current branch. If multiple branches are labeled, we find the slope angle of each branch (using endpoints), and the branch with the smallest slope difference is selected to label the current branch. If none of the associated branches are labeled, we wait for at least one to get assigned. Iteratively we label all the branches. As a result, we would have a part-labeled skeleton.

3.5.3 Skeletonized Superpixel Propagation

After labeling all branches as shown in Figure 3.5 box D, we propagate the labels to all co-segmentation mask pixels. To ensure natural boundaries, computational efficiency, and reduced complexity, we use the Simple Linear Iterative Clustering (SLIC) [101] algorithm to convert the original image into superpixels. Each superpixel has its centroid coordinates and the mean color of its pixels. For the new seed-labeled branches, we identify the centroid of each label referred to as part centroid shown in Figure 3.5 box D.

We calculate the distance between each superpixel centroid and part centroid in

the (R, G, B, x, y) dimension. The label of the closest part centroid is assigned to the current superpixel. This process is repeated for all superpixels, assigning all superpixels the seed labels(where each seed corresponds to a specific object part).

$$sd_k \;=\; \sqrt{(sp_x - pc_{kx})^2 + (sp_y - pc_{ky})^2} \tag{3.6}$$

$$cd_k \;=\; \sqrt{(sp_R - pc_{kR})^2 + (sp_G - pc_{kG})^2 + (sp_B - pc_{kB})^2} \tag{3.7}$$

$$dist_k \;=\; \lambda \cdot sd_k \;+\; (1 - \lambda) \cdot cd_k \tag{3.8}$$

$$label \;=\; \arg\min_i \left(dist_k\right) \;, \forall_{i \in 1 \ldots k} \tag{3.9}$$

In the equation 3.6 and 3.7, sp refers to a superpixel. The sp_x, sp_y, sp_R, sp_G, and sp_B refer to the x, y coordinate and RGB magnitude of the superpixel centroid respectively. Similarily, pc_{kx}, pc_{ky}, pc_{kR}, pc_{kG}, and pc_{kB} refer to the x, y coordinate and RGB magnitude of the k-th part centroid respectively.

We use equation 3.6 and 3.7 to calculate the spatial and color distance of a superpixel with the k-th part centroid. The final distance is calculated with the equation 3.8, where λ balances the color and spatial distances to determine which object part a superpixel should be assigned. As shown in equation 3.9, the superpixel is assigned the label of part-centroid whose aggregate distance is minimum. The resulting image is shown in Figure 3.5 box D under object co-part segmentation. The part propagation process is applied to all images through skeletonized superpixels.

We will see the process of skeletonized superpixel propagation through an example. As seen in the figure 3.7, using the SLIC algorithm, we generate superpixels. Now some of these superpixels are of background and some are of object

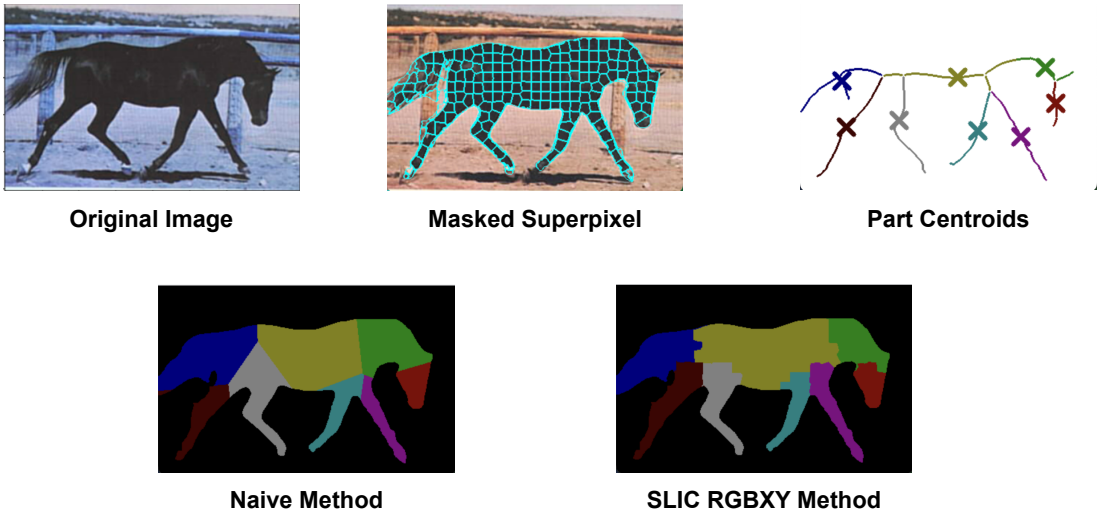

Figure 3.8: Different stages of superpixel propagation. The masked SLIC superpixels generated from original image, and part centroids are compared to generate part segmentation. We compare the superpixel based method with a naive method. The naive method is based on matching pixels directly with part centroids without forming any superpixels.

region. Few of them cover the object region as well as the background. We segment out the superpixels that only cover the object region, and then segment the object region coordinates from the superpixels that cover both the background and object. The final superpixels are marked under the 'masked Superpixels'. These are the superpixels that we are interested in dealing with. We find the mean color and centroid of only these superpixels.

The figure 3.8 is a continuation of the figure 3.7. The masked superpixels generated in the figure 3.7 are compared with the part centroid to obtain labelled parts. We compare it with other processes which do not use superpixel propagation. In 'Naive Method', we do not create any superpixels, and directly compare each pixel of object region with part centroid. We can see that the leg region encroaches heavily in the torso region and the boundaries are unnatural. In the 'SLIC RGBXY Method', we compare the superpixels of the object region with the part centroids, and see that the boundaries are more natural and dependent on the size of superpixels which can be controlled. Also the use of mean color and mean distance makes sure that the torso region is not unjustifiable encroached

by the leg region.

3.6 Implementation details

We propose an end-to-end pipeline to extract semantic parts from a set of similar images. Our proposed method does not involve a training phase. In case of a large number of images, we cluster them based on image features and apply our method to all the resulting clusters. The image whose features are nearest to the cluster center serves as the key image, and we share it with the user for interaction. In order to tackle a large number of images of same class, we cluster them into different groups. The methodology explained in chapter 3 is applied to each cluster separately.

We perform the clustering using k-means algorithm using image features obtained from any state-of-the-art deep learning architecture. We also try the GIST features [102] which are known to be good for clustering based on orientation and pose.

This way, we have d dimensional feature vector F_i to represent the i-th image of the j-th cluster with k images. Then these features are clustered with the help of k-means algorithm.

$$R_j \;=\; \arg\min_{i} \; d_{eucd}(F_i,\; center) \;\;, \forall_{i \in 1 \ldots k} \tag{3.10}$$

In equation (3.10), d_{eucd} calculates the euclidean distance between two vector representations. In equation (3.10), R_j is representative image of j-th cluster. There are k images in each cluster where k is varying depending on size of cluster. The image with representation F_i nearest to the cluster center is termed as cluster representative.

For each cluster, the user is shown its representative image and requested to provide the seeds for the parts that are visible to them. We propagate these seed cue to all the images of the same cluster. We propagate seed cues for each cluster. We prefer the point correspondence (section 3.4.2) for seed propagation over deep

learning based dense correspondence (section 3.4.2). Empirically we have seen that the point correspondence (section 3.4.2) performs better. The results can be seen in Table 4.6 of ablation study (section 4.5). Once we have the seed cue on each image, we match them with the branches of the segmented skeleton mask, which are then propagated to the object mask.

Chapter 4

Experimental Results

4.1 Dataset

To verify its effectiveness, we have evaluated the proposed approach on the seven major classes of the PASCAL 2010 part dataset [1]- Horse, Cow, Bird, Cat, Dog, Person, and Sheep. However, as the ground truth object annotation is highly detailed, we merged the segments that refer to similar parts into a new unique part. For instance, we merged the right front leg and right front paw into the right front leg.

As we can see in the figure 4.1, the original groundtruth is very detailed. For the boy, the eyebrows, and eyes are separate. We combine all of the facial features along with hair under a common label referred to as head. For human and animal, the high level part understanding is used such as head, torso, right leg, left leg, tail etc. The PASCAL part annotations [103] shared here are for the train and validation set. We extract the images from the train and validation which contains the single object from one of the seven classes selected. We, also, filter out the images which have a bounding box area of less than 20% as the objects with such small region are not very detailed to carry out part segmentation. The statistics of the dataset are in table 4.1. Out of 1948 images, 947 images are from training set and 1001 images are from the validation set. We show the combined results on the training and validation set(trainval) and validation set separately.

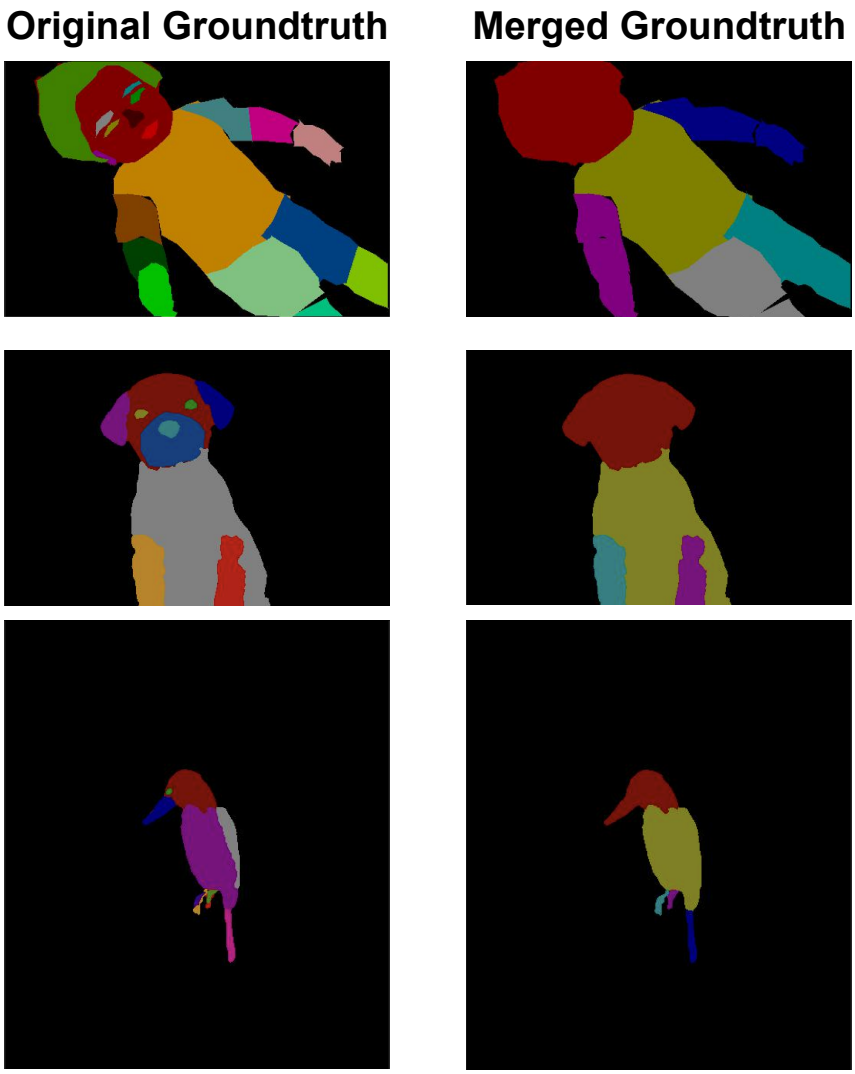

Figure 4.1: Dataset: Three examples of the original ground truth annotation and merged ground truth used in the work. We have created simplified ground truth by merging detailed parts(such as eyes, ear, mouth, hair etc.) in to a larger unified part(head).

4.2 Evaluation Metric

We use Jaccard Similarity to evaluate the segmentation masks of the images and a modified form of Intersection over Union(IOU) is used to evaluate the semantic part segmentation results of PASCAL 2010 part dataset [1].

Intersection over Union(IOU) is defined as

$$IOU(GT, OP) = \frac{GT \cap OP}{GT \cup OP} \tag{4.1}$$

Class	Training	Validation
Bird	133	121
Cat	303	316
Cow	35	35
Dog	264	266
Horse	48	60
Person	131	176
Sheep	33	27
Total	947	1001

Table 4.1: Total number of images in each class. We show the total images in training set and validation set separately.

where OP is the segmentation output/results and GT is the ground truth region. For a single image the pixels for which both the predicted and groundtruth region have same labels are counted as intersection, and the union is the the sum of counts of pixels for the object region in both images with intersection count reduced from it. We get a ratio of the overlapping region of both images with the total region covered by either image.

We discuss in detail the modified version of IOU used for part evaluation. Given that we have n images $I = \{I_1, I_2, ..., I_n\}$. We take average IOU for verification which is defined as

$$IOU_a(G, P) \;=\; \frac{1}{m}\sum_{j=1}^{m}\max_{k}\frac{1}{n}\sum_{i=1}^{n} IOU(g_{ij}, p_{ik}) \tag{4.2}$$

In equation (4.2), $G = \{G_1, G_2, ..., G_n\}$ is set of ground truth parts and $P = \{P_1, P_2, ..., P_n\}$ is set of part segmentation results for image set I. There are max m unique parts in the ground truth. The g_{ij} is the mask of j-th ground truth part of the i-th image, and p_{ik} is the mask of k-th segmentation result part of the i-th image.

For j-th ground truth part of the image set, we go through each output part present in its corresponding prediction images. The output part(k) with max IOU value(as defined by (4.1)) is chosen as corresponding output part for j-th ground truth part. While deciding what should be the max IOU value, we find the IOU of the j-th ground truth part with k-th output part for all the n images.

For each ground truth part, we find max IOU corresponding to it and sum them. We take average over number of ground truth parts.

The above evaluation process is for a set of images. Now as we saw in the implementation details, all the images of a class are clustered. We use equation (4.2) to find IOU of each cluster. The final IOU presented is the average of the number of clusters for each class. There are two ways we calculate the final IOU for different classes for comparison with the other methods. First way is to normally iterate over each cluster, calculate its IOU and then report the average IOU. This methods takes into account the representative/key image of the cluster i.e. the image user interacted with. However, when we compare with the methods which do not receive supervision in any way such as unsupervised methods, we deploy another way of calculating the IOU to keep the comparison fair. This method removes the key image from the cluster before evaluating the IOU. The act of removing the key image makes sure that any image that has received direct user interaction is out of the evaluation process. We use these two methods depending on what we are comparing with. If we are comparing with weakly supervised approaches, we keep account of key/supervised image and if we are comparing with unsupervised approaches, we remove the key/supervised image from the cluster.

4.3 Qualitative Results

In figure 4.2, we have presented two images of each class in two panels to visualize the results of our approach. This version of the results is with GluNet [10] based seed propagation and branch labelling without unsupervised mask. The results of co-segmentation can detect the object region properly other than for the leg region of a few images such as left panel dog and right panel horse where the left and right back leg are combined as one because of less space between them. The result of skeletonization can detect object outline well and present an idea of what the object would look like in skeleton form. We have drawn seed cues on the

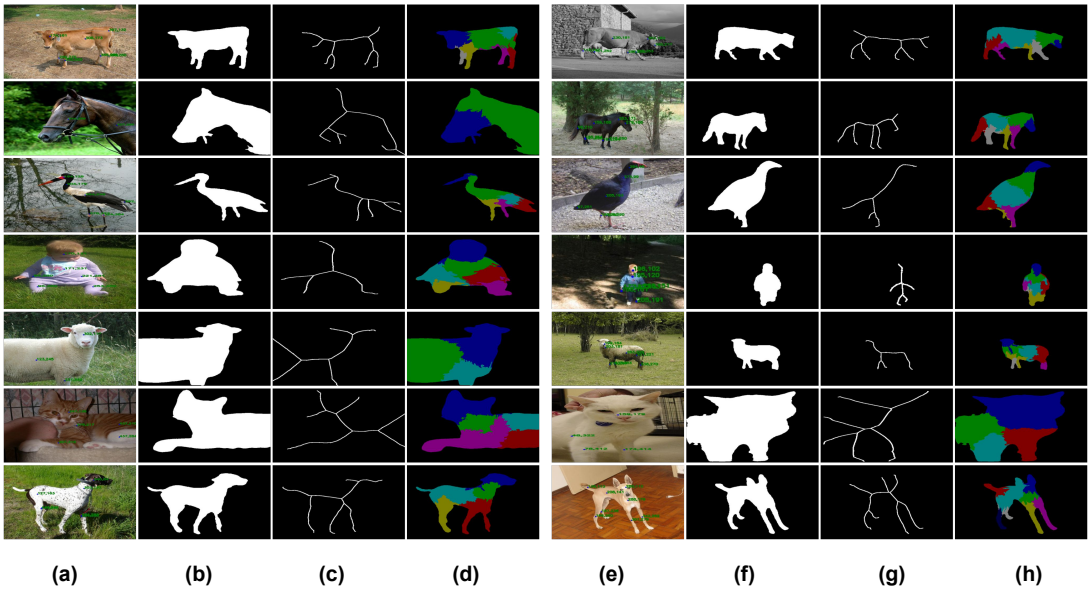

Figure 4.2: Part Segmentation Results: (a) and (e): original image with seed cue , (b) and (f): object segmentation mask , (c) and (g): object skeleton mask, and (d) and (h): part segmentation results

original image, and we can see that clustering and seed propagation methodology can propagate seeds well to most parts of the object. The two legs of the dog in the left panel, the tail of a cow in the right panel, the left back leg of a horse in the right panel, and the right leg of the child in the right panel are some of the missed cases which seed propagation was not able to identify. As visualized in Figure 4.2 for PASCAL Part, the part segmentation approach can identify most parts correctly, treating a seed cue as a single part. Our part segmentation results in Figure 4.2 (d) and (h) are highly dependent on the seed location and the skeleton outline. In the child image in the left panel, the skeleton mask cannot recognize the child's left hand because of which part segmentation approach identifies part of the torso as the left hand. For the cat in the left panel, the front leg seed is also propagated to part of the torso because the leg branch in the skeleton mask extends to the torso region. For the dog object in the left panel and the horse object in the right panel, both the back leg are identified as one part because of a single seed on the original image, and the skeleton outlining both legs as one because of the joint object segmentation mask.

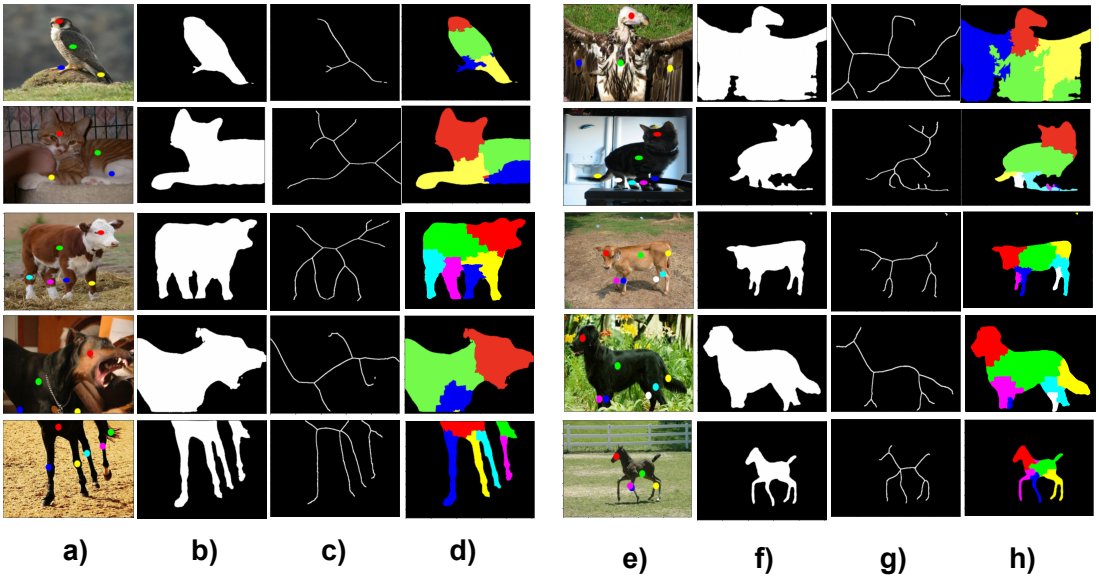

Figure 4.3: Part Segmentation Results: (a) and (e): original image with seed cue , (b) and (f): object segmentation mask , (c) and (g): object skeleton mask, and (d) and (h): part segmentation results

In Figure 4.3, we present part segmentation results along with the original image, seed cue, co-segmentation, and skeleton mask. These examples illustrate the effectiveness of our proposed method with ViT based point correspondence and branch labelling with help of unsupervised mask. By drawing seed cues on the original image, we can observe that the seed propagation method can update seeds effectively to most parts of the object. The constraint in the part segmentation due to the object region is observed here as well. For example, the cat in the right panel has well placed seed cues but due to the error in object region segmentation, we are unable to label the left front leg. The distortion in object region causes the skeleton to change its shape and the three of the legs are merged in one branch which makes it impossible to segment all the legs properly. The right panel cow, suffers from the same issue as well. The tail region is not separately identifiable from the torso region because of which the seed cue of the tail region takes over the region of torso when labelling is done. The issue is more prevalent when two or more part so close to each other that there object region is one connected component so it becomes really tough to label them distinctly. The horse in the right panel is a good example of how the error in seed cue

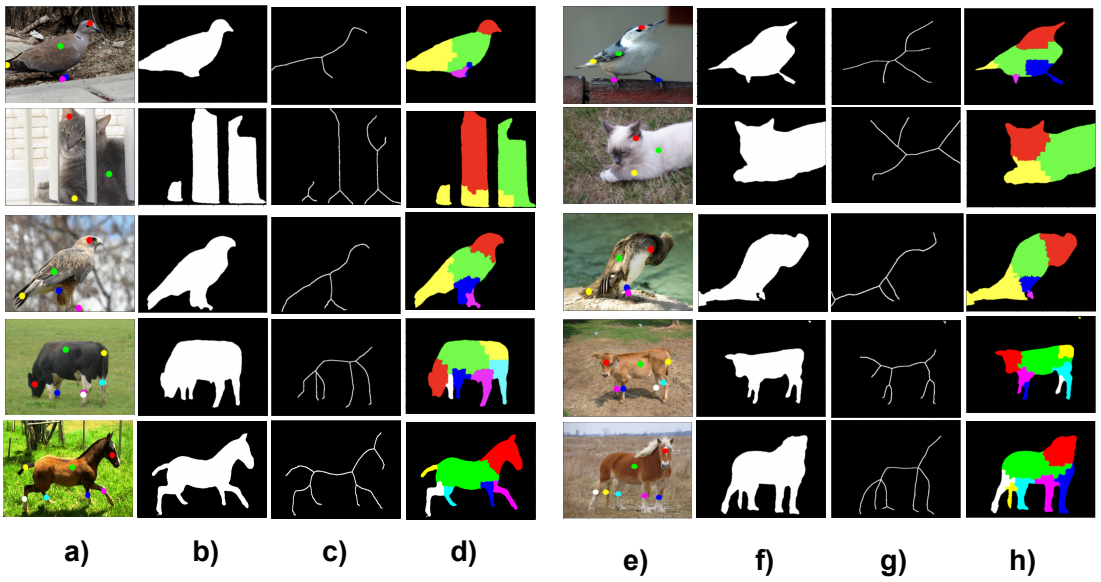

Figure 4.4: Part Segmentation Results: (a) and (e): original image with seed cue , (b) and (f): object segmentation mask , (c) and (g): object skeleton mask, and (d) and (h): part segmentation results

propagation is corrected by the branch labelling process. There are two seed on the left front leg because of error in seed cue propagation but it is corrected by the labelling process through which the second seed is placed to its proper position.

In Figure 4.4, we present few more results to show how our method is able to produce good segmentation results. The bird in the first row in the right panel has a part of its leg in the torso which can be attributed to the shape geometry of its skeleton and the longer branch instance. The cat in the left panel is a good example how we are overcome the obstructions well placed cues and efficient branch labelling process. The horse in the last panel are able to properly segment the parts.

The visualisation of the results helps us see where our method is able to perform well and where it lacks. We can observe that the reliance on object region can be seen in the skeletonization shape, and then further into how branches are labelled. In the next section, we see the quantitative comparison of our method with other weakly supervised, interactive and unsupervised methods.

	Bird	Cat	Cow	Dog	Horse	Person	Sheep	Average
trainval(with supervised)	0.502	0.463	0.521	0.468	0.466	0.444	0.592	0.494
trainval(without supervised)	0.377	0.387	0.490	0.389	0.415	0.390	0.552	0.428
val(with supervised)	0.547	0.481	0.539	0.469	0.496	0.468	0.581	0.511
val(without supervised)	0.371	0.401	0.453	0.357	0.429	0.414	0.378	0.400

Table 4.2: The objective (IOU score) on our technique for trainval(both train and validation images) and val(validation alone) is shown with both approaches- with and without supervised images.

4.4 Quantitative Results

Two different approaches are used to calculate the modified Intersection over Union (IOU) score. The first approach considered all the images in the cluster, including the key image, while the second approach excluded the key image to make the process fairer for unsupervised methods.

In table 4.2, we show the IOU score of our method using equation (4.2). The score is calculated on the trainval set and val(validation) set separately. This is done to compare the performance of our method if only the validation set is considered. We calculate the IOU using both the approaches - taking supervised/key images into account and without supervised/key images.

Our results are compared with weakly supervised approaches that involve some form of user interaction, including methods such as [12, 13, 14, 15, 9]. We account for all images for the comparison. We performed better than the recent user-interaction-based method [9] (Refer to table 4.3), and also outperformed [16].

When comparing with methods which are unsupervised in nature, we want to maintain the comparison fair as our images are weakly supervised while unsupervised method have image-level label and number of parts to generate. We remove the images the user supervised and only calculate IOU for the image which have the seeds propagated through our method. We calculate the IOU for the same images through unsupervised method as well. The unsupervised method generate parts based on number of parts instructed before running the method, which makes them constrained to produce set number of parts. To overcome this constraint, we share the inputs provided to our method by user with unsuper-

	Bird	Cat	Cow	Dog	Horse	Person	Sheep	Average
Meng *et al.* [11]	0.111	0.113	0.124	0.142	0.075	0.128	0.106	0.114
Meng *et al.* [12]	0.064	0.053	0.049	0.047	0.026	0.032	0.043	0.044
Guillaumin *et al.* [13]	0.131	0.055	0.101	0.120	0.103	0.083	0.115	0.101
Dong *et al.* [14]	0.118	0.120	0.097	0.069	0.089	0.039	0.093	0.089
Krause *et al.* [15]	0.099	0.135	0.115	0.141	0.067	0.106	0.105	0.109
Meng *et al.* [9]	0.330	0.327	0.224	0.409	0.293	0.264	0.362	0.315
DINO-ViT(with supervised images) [16]	0.456	0.425	0.476	0.451	0.459	0.353	0.579	0.457
Ours(with supervised images)	0.502	0.463	0.521	0.468	0.466	0.444	0.592	0.494

Table 4.3: The objective (IOU score) on our technique and its comparison with other weakly supervised techniques [11, 12, 13, 14, 15, 9] and unsupervised method [16] taking into account supervised images. Red refers to highest score and Blue refer to second highest score.

vised method. For each cluster in unsupervised, we segment the object into the same number of part as the seeds given by the user to our method. This way we are able to remove the constraints of fixed parts by making sure unsupervised method also has access to users insight on how many visible parts are there in the key image/object. Also with DINO-ViT[16], we introduce supervision during its part-clustering process. This is done by manipulating the clustering process of the [16]. The seeds that we have for each image/object are shared with [16] method. And during each clustering process, these seed coordinates are added to their randomly sampled points. At the end, there is a clustering step to identify the part label of each image patch. In this step, we make our key image seeds cues/coordinates as the initial centroid. Instead of a random centroid initialization, we give the method a head start on where the parts might be located so it can move in the rigth direction. To summarize, we give unsupervised methods a fair chance during evaluation process by sharing the weakly supervised insights with them in following form - 1) Removed the key images which are supervised by user during evaluation process , 2) Vary the number of parts to be generated using number of seeds identified by user, and 3) Use the seed location during the clustering process of [16] to guide the method in the right direction.

The results of DFF [17] and DINO-ViT [16] (with the external supervision mentioned in above paragraph and without any supervision i.e. vanilla model

	Bird	Cat	Cow	Dog	Horse	Person	Sheep	Average
DFF [17]	0.199	0.227	0.249	0.247	0.229	0.181	0.379	0.244
DINO-ViT(w/o external supervision) [16]	0.308	0.3428	0.449	0.351	0.440	0.295	0.562	0.392
DINO-ViT [16]	0.360	0.383	0.457	0.392	0.399	0.305	0.546	0.406
Ours	0.377	0.387	0.490	0.389	0.415	0.390	0.552	0.428

Table 4.4: The objective (IOU score) on our technique and its comparison with unsupervised techniques [17, 16]. The score is calculated after removal of users supervised images. Red refers to highest score and Blue refer to second highest score.

)are presented in table 4.4. We can see that our method is able to produce higher quality part segmentation.

4.5 Ablation Study

In this section, we analyze the influence of the three key components presented in the proposed method. We check if we remove any of the part, how will it impact the evaluation and the quality of part segmentation. To investigate this, we conducted three variations of the method, which included:

1. A version without the finer skeleton decomposition using an unsupervised mask i.e, we just decompose the original skeleton into branches depending on finding the convergence point where multiple branches meet. This can lead to cases where the branches are unnecessary long and encroach the region of another part. So theoretically the result should be inferior to the finer branch decomposition.

2. A version without the sophisticated branch labeling (instead using naive branch-seed labeling). In the naive labeling, we take the set of branch centroids and match them with the seeds. The branch centroid closest to the seed is given its label. Each branch is assigned some part-label depending on the seed they are closest to. This modeling method can miss out few seeds if they are always second in the distance with some other seed. This method naively matches seeds to the branch without taking into account the seed positioning on the key image, the relative seed positioning in current

	Bird	Cat	Cow	Dog	Horse	Person	Sheep	Average
W/o finer skeleton decomposition	0.468	0.446	0.508	0.431	0.447	0.431	0.513	0.463
W/o unsupervised branch labeling	0.457	0.429	0.481	0.424	0.433	0.407	0.493	0.446
W/o skeletonized superpixel	0.413	0.394	0.460	0.395	0.372	0.366	0.497	0.413
All components	0.502	0.463	0.521	0.468	0.466	0.444	0.592	0.494

Table 4.5: The objective (IOU score) on our technique accounting the supervised images and its comparison with different modules. Red refers to highest score and Blue refer to second highest score.

	Bird	Cat	Cow	Dog	Horse	Person	Sheep	Average
Dense Correspondence(with supervised)	0.468	0.44	0.503	0.441	0.477	0.431	0.573	0.476
Point Correspondence(with supervised)	0.502	0.463	0.521	0.468	0.466	0.444	0.592	0.494
Dense Correspondence(without supervised)	0.339	0.360	0.477	0.345	0.437	0.367	0.518	0.406
Point Correspondence(without supervised)	0.377	0.387	0.490	0.389	0.415	0.390	0.552	0.428

Table 4.6: The objective (IOU score) on our technique for trainval(both train and validation images)shown with both seed propagation approaches(section 3.4) for - with and without supervised images evaluation technique.

image, the unsupervised region of both images and the positioning of the seed cue in unsupervised mask regions. Theoretically this should be inferior to using a sophsticated branch labelling process.

3. A version that substituted the use of skeletonized superpixels with naive pixel-branch centroid matching for part label propagation to the object region. The naive approach here refers to iterating over each pixel location of the object region and checking which part-centroid is nearest to it, and assigning the same part-label to the object region pixel.

The results, as demonstrated in Table 4.5, clearly indicate that the omission of any of these components resulted in a decline in the overall accuracy of the method.

We compare the two different approaches for seed propagation discussed in the section 3.4. The results can be seen in the Table 4.6. We test the approach on the complete trainval data and see that for both evaluation technique(accounting for supervised images and without supervised images), point correspondence approach performs better. We prefer the point correspondence (section 3.4.2) for seed propagation over deep learning based dense correspondence (section 3.4.2).

Chapter 5

Unsupervised Part Segmentation

5.1 Introduction

We have seen so far that our work for skeleton-based interactive co-part segmentation is able to produce better qualitative and quantitative results then the contemporary methods. When going through the literature, we saw in section 2.4.2 that there are many unsupervised techniques for part segmentation. The methods [7, 26, 6] propose a deep learning architecture for unsupervised part generation. The model learns based on the loss functions. These loss functions are based on the properties of natural parts, such as connectivity, all parts form object etc. There are three main drawbacks of the these models- number of parts have to be predefined, a new model needs to be trained anytime we are changing the number of predefined parts or object class, and the parts formed are not meaningful in nature.

We have introduced an unsupervised method which deals with the slow and complex training drawback and aims to produce parts which are meaningful in nature. The user still shares the number of predefined parts to segment, however every time the number of parts are changed, we don't have to go through a long training process.

5.2 Proposed Methodology

5.2.1 Method Overview

In the unsupervised approach, we do not share information between different images. The part generation is done with the help of the co-segmentation mask

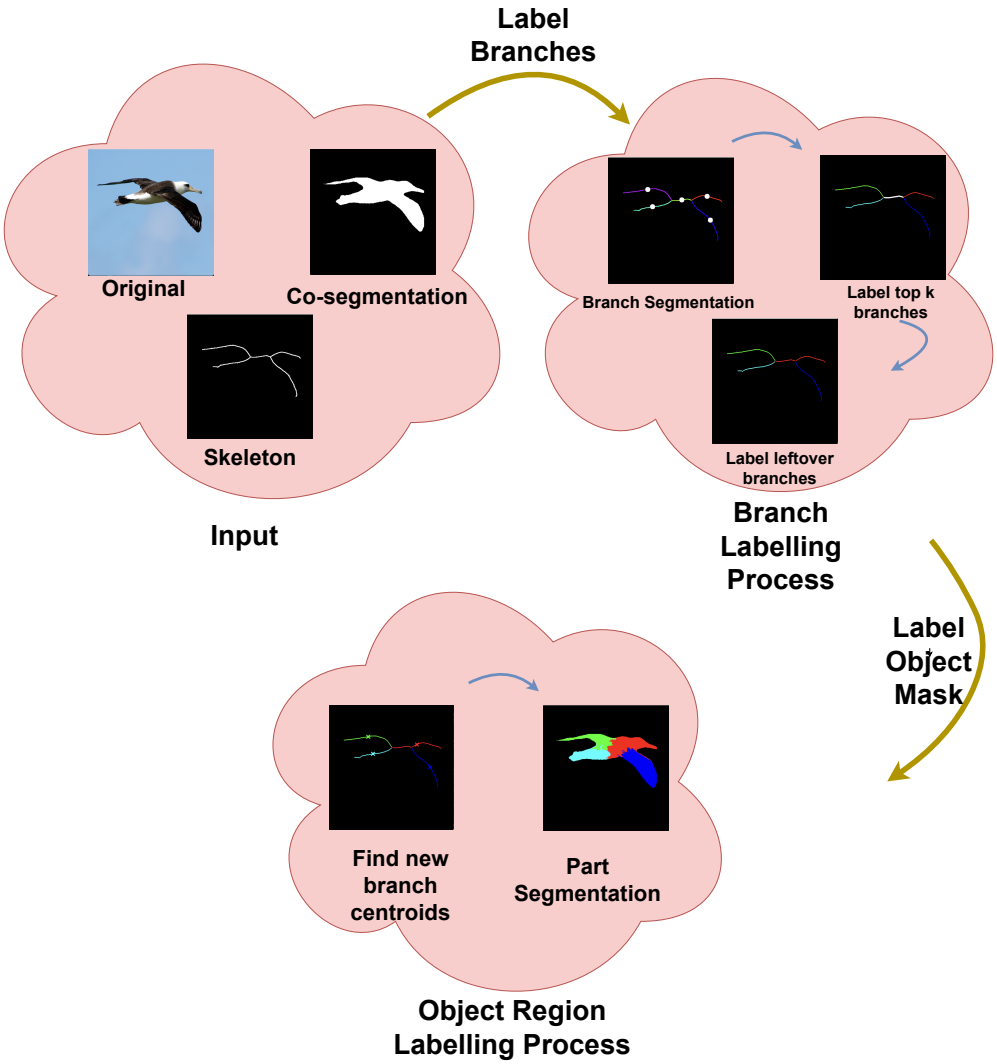

Figure 5.1: Flowchart of the Unsupervised approach. The input is an image, its object mask and its skeleton. In the first process, branches are segmented, longest k branches are uniquely labeled, and then rest of branches are labeled. The branch labels are then propagated to object region.

and the skeleton of the object. We iterate over each image and perform part segmentation with object, skeleton mask and number of predefined parts as seen in Figure 5.1. We can see that there are two main parts of the method - How do we label skeleton branches and how is the branch label propagated to the object region. We answer both of these question in next sections.

5.2.2 Branch Labeling

At first, the object skeleton is segmented into branched based on where the there is junction point. As seen in the figure 5.1, the skeleton here is segmented into

five branches. Lets say the number of parts to segment is k, we find the top k branches based on their size. Each of these branches is labeled with a unique part label. The idea here is that the longest branches would for the basis of a part and smaller branches will then be matched to one of these parts.

Once we have labeled top k branches with a part label as seen in Branch Labelling Process of figure 5.1, we look at leftover branches. The branches which have a single neighbouring branch is labeled with its connected branches label. If there are multiple neighbours, we find the slope angle of each neighbouring branch and the one with the closest angle is chosen to label. This way we label all the leftover branches.

5.2.3 Object Region Propagation

As seen in the Object Region Labelling Process of figure 5.1, we find the new part centroids. Each part centroid is the center of the branches labeled with that part. These part centroids are matched with the superpixels of the original image in the same way as done in section 3.5.3. We follow the same process as in section 3.5.3 to propagate the part label to object region with the help of the superpixels. Finally part segmentation of an object is achieved as seen in the figure 5.1.

We iterate over all the images in a class and perform the same methodology to achieve the part segmentation.

5.3 Dataset

We work with the CUB dataset [104]. CUB dataset has 200 classes with part location annotation for 15 parts. Unlike PASCAL part dataset [1] which has pixel-level label for each part, CUB dataset [104] has centroid coordinate of each part referred to as part location. We choose the first three classses of CUB dataset for experimentation to maintain consistency with the results of [7]. The classes are - CUB-001(Black Footed Albatross), CUB-002(Laysan Albatross), and CUB-003(Sooty Albatross). Each class has 60 images.

5.4 Evaluation Metric

We cannot use the Intersection over Union as shown in equation (4.2) because the CUB dataset has no pixel-level ground truth. Instead, we use the evaluation method used by state-of-the-art unsupervised methods [7, 26, 6]. We convert each part result into landmark by taking part centers and evaluate against groundtruth annotations. We use the method in [7] to fit a linear regressor that learns to map the detected landmarks to groundtruth landmarks. We report the landmark regression error in terms of mean L2 distance. To account for varying bird sizes, we normalize by the bounding box as done in other methods. We train the linear regressor on the training images(30) and test the method on rest of the images.

5.5 Experimental Results

We can see in the figure 5.2 the visual results of the unsupervised method when compare to the state of the art unsupervised methods such as SCOPS [7] and Choudhury *et al.* [6]. We can see in the results of the SCOPS [7] that the parts are not well defined. Left wing part and Right wing part is separable easily. On the other hand, Choudhury *et al.* [6] performs better on part classification of the left and the right wing. Our unsupervised method shows that the parts are well defined however there is a lack of boundary or sense of where part end. This is because the part labelling is entirely dependent on the skeleton structure and no other input is used. Which makes it harder to evaluate what a part is just based on the skeleton branches.

We share the quantitative comparison of our unsupervised method with state-of-the-art unsupervised methods in Table 5.1. The results are obtained on the first three classes of the CUB dataset [104] for four predefined parts. This means that all the method try to segment the object into four different parts. Using the evaluation metric described in the 5.4, we calculate the L2 error of all three classes.

We can see that our unsupervised model performs worse than the contempo-

Figure 5.2: Visual Results on CUB: (a): original image, (b): SCOPS [7] , (c): Choudhury *et al.* [6], and (d): Unsupervised(ours)

rary methods. The L2 distance error is the highest when compared to the other methods.

	CUB-2011 Dataset [104]			
	CUB-001	CUB-002	CUB-003	Average
Zhang *et al.* [105]	30.12	29.36	28.19	29.22
DFF [17]	22.42	21.62	21.98	22.00
SCOPS [7]	18.50	18.82	21.07	19.46
Liu *et al.* [26]	18.15	17.54	19.40	18.36
Unsupervised(ours)	28.3	22.8	39.2	30.1

Table 5.1: The landmark evaluation on CUB datastet. Normalized L2 distance comparing our approach to recent techniques(K=4)

5.6 Observation

In this section, we discuss the reason for the poor performance of the unsupervised method proposed by us. We are assigning the K longest branches in terms of size as the first K parts. The assumption is that the largest branches would be from different reason and would present a part. But this assumption is naive and is not good enough to define what a part is. There is no sense of ordering in the parts which is the biggest reason for the poor performance on the evaluation metric. The largest branch in one object might correspond to the head while in other object, it may correspond to leg region. If we are assigning label 1 to the longest branch, the meaning of the label keeps on changing which goes against the notion of what part is. A part should represent consistent region across objects of same class. The L2 error shows the measure of how consistent are part in their location with respect to the part locations in ground truth. In our method we lose that consistency because we are unable to define which branch should represent which part.

Chapter 6

Concluding Remarks

6.1 Conclusion

This paper presents a novel approach for interactive co-part segmentation, which aims to accurately segment the semantic parts of objects in a collection of similar images through co-skeletonization. The proposed method involves interaction with a key image to generate seed cues, followed by seed propagation to update the seeds in the remaining images. The part propagation is achieved by matching the seeds with the segmented branches of the object skeleton and propagating the branch label to the object region. The effectiveness of the proposed method is evaluated on the seven major classes of PASCAL Part dataset. Our approach outperforms both interactive and unsupervised methods and produces semantically meaningful parts. Our results demonstrate the efficacy of the proposed interactive co-part segmentation method in achieving high-quality part segmentation.

6.2 Future Direction

Our interactive co-part segmentation can be improved in few ways. We have used the co-segmentation approach to extract the segmentation mask, which requires the class label of each image to be known in advance and the object to be salient throughout the image set. If the object is not salient, the object mask suffers. The skeletonization provides an outline that is highly dependent on the object mask shape, and if the object mask is not able to give an accurate distinct shape which can help identify each part separately, the skeleton will merge different part branches into one. We request the user's seed cue on only key/representative

images and propagate the seed cue on the rest of the images in the group. This can lead to a misplaced seed cue if the image in the group varies significantly from the representative image in shape and pose. It could also lead to a case where the representative image has a small number of visible parts, and other images in the group have more parts for which we do not have a seed cue. For example, the interactive seed cue process is on a horse image where only the head, neck, and torso are visible; thus user provides three seed cues. We try to extend the seeds to another horse image in the group, with legs and tail visible along with the above parts. In such a case, we encounter an issue where it is impossible to have any seed cue for the tail and legs. We can tackle these issues in the future work.

We can take up other directions for part segmentation. There are a lot of new methods which are working on the task of referring image segmentation. The referred image segmentation [106, 107, 108] is the labeling of pixels of an image to extract a particular object referred to by the linguistic expression. For example, *the person with the blue scarf* should only segment the person wearing a blue scarf from the surrounding. We can leverage such models and improve them so they are able to identify the distinct semantic parts and can refer to them. The expression *The head of the cat* should be able to segmentation the head region of the cat.

REFERENCES

[1] 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*, 2014, pp. 1971–1978.

[2] Y. Su, J. Deng, R. Sun, G. Lin, and Q. Wu, "A unified transformer framework for group-based segmentation: Co-segmentation, co-saliency detection and video salient object detection," *arXiv preprint arXiv:2203.04708*, 2022.

[3] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns," *Communications of the ACM*, vol. 27, no. 3, pp. 236–239, 1984.

[4] T. Taniai, S. N. Sinha, and Y. Sato, "Joint recovery of dense correspondence and cosegmentation in two images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4246–4255.

[5] J. Wang and A. L. Yuille, "Semantic part segmentation using compositional model combining shape and appearance," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1788–1797.

[6] S. Choudhury, I. Laina, C. Rupprecht, and A. Vedaldi, "Unsupervised part discovery from contrastive reconstruction," *Advances in Neural Information Processing Systems*, vol. 34, pp. 28 104–28 118, 2021.

[7] W.-C. Hung, V. Jampani, S. Liu, P. Molchanov, M.-H. Yang, and J. Kautz, "Scops: Self-supervised co-part segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 869–878.

[8] H.-S. Fang, G. Lu, X. Fang, J. Xie, Y.-W. Tai, and C. Lu, "Weakly and semi supervised human body part parsing via pose-guided knowledge transfer," *arXiv preprint arXiv:1805.04310*, 2018.

[9] F. Meng, H. Li, Q. Wu, K. N. Ngan, and J. Cai, "Seeds-based part segmentation by seeds propagation and region convexity decomposition," *IEEE Transactions on Multimedia*, vol. 20, no. 2, pp. 310–322, 2017.

[10] P. Truong, M. Danelljan, and R. Timofte, "Glu-net: Global-local universal network for dense flow and correspondences," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 6258–6268.

[11] F. Meng, H. Li, Q. Wu, B. Luo, and K. N. Ngan, "Weakly supervised part proposal segmentation from multiple images," *IEEE Transactions on Image Processing*, vol. 26, no. 8, pp. 4019–4031, 2017.

[12] F. Meng, H. Li, G. Liu, and K. N. Ngan, "Object co-segmentation based on shortest path algorithm and saliency model," *IEEE transactions on multimedia*, vol. 14, no. 5, pp. 1429–1441, 2012.

[13] M. Guillaumin, D. Küttel, and V. Ferrari, "Imagenet auto-annotation with segmentation propagation," *International Journal of Computer Vision*, vol. 110, no. 3, pp. 328–348, 2014.

[14] X. Dong, J. Shen, L. Shao, and M.-H. Yang, "Interactive cosegmentation using global and local energy optimization," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3966–3977, 2015.

[15] J. Krause, H. Jin, J. Yang, and L. Fei-Fei, "Fine-grained recognition without part annotations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5546–5555.

[16] S. Amir, Y. Gandelsman, S. Bagon, and T. Dekel, "Deep vit features as dense visual descriptors," *arXiv preprint arXiv:2112.05814*, vol. 2, no. 3, p. 4, 2021.

[17] E. Collins, R. Achanta, and S. Susstrunk, "Deep feature factorization for concept discovery," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 336–352.

[18] J. Krause, T. Gebru, J. Deng, L.-J. Li, and L. Fei-Fei, "Learning features and parts for fine-grained recognition," in *2014 22nd International Conference on Pattern Recognition*. IEEE, 2014, pp. 26–33.

[19] F. Xia, P. Wang, X. Chen, and A. L. Yuille, "Joint multi-person pose estimation and semantic part segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6769–6778.

[20] H. Huang, W. Yang, J. Lin, G. Huang, J. Xu, G. Wang, X. Chen, and K. Huang, "Improve person re-identification with part awareness learning," *IEEE Transactions on Image Processing*, vol. 29, pp. 7468–7481, 2020.

[21] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, "Person re-identification by multi-channel parts-based cnn with improved triplet loss function," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1335–1344.

[22] Z. Zhou, F. Shi, and W. Wu, "Learning spatial and temporal extents of human actions for action detection," *IEEE Transactions on multimedia*, vol. 17, no. 4, pp. 512–525, 2015.

[23] P. Luo, X. Wang, and X. Tang, "Pedestrian parsing via deep decompositional network," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[24] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. L. Yuille, "Joint object and part segmentation using deep learned potentials," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[25] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, "Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline)," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[26] S. Liu, L. Zhang, X. Yang, H. Su, and J. Zhu, "Unsupervised part segmentation through disentangling appearance and shape," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8355–8364.

[27] Q. Gao, B. Wang, L. Liu, and B. Chen, "Unsupervised co-part segmentation through assembly," in *International Conference on Machine Learning*. PMLR, 2021, pp. 3576–3586.

[28] W. Yang, P. Luo, and L. Lin, "Clothing co-parsing by joint image segmentation and labeling," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 3182–3189.

[29] S. Naha, Q. Xiao, P. Banik, M. A. Reza, and D. J. Crandall, "Pose-guided knowledge transfer for object part segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 906–907.

[30] N. Tritrong, P. Rewatbowornwong, and S. Suwajanakorn, "Repurposing gans for one-shot semantic part segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 4475–4485.

[31] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:1412.7062*, 2014.

[32] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[33] N. Dhanachandra, K. Manglem, and Y. J. Chanu, "Image segmentation using k-means clustering algorithm and subtractive clustering algorithm," *Procedia Computer Science*, vol. 54, pp. 764–771, 2015.

[34] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[35] N. Plath, M. Toussaint, and S. Nakajima, "Multi-class image segmentation using conditional random fields and global classification," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 817–824.

[36] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[37] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[38] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[39] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[40] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[41] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[42] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.

[43] W. Li, O. Hosseini Jafari, and C. Rother, "Deep object co-segmentation," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 638–653.

[44] K.-J. Hsu, Y.-Y. Lin, and Y.-Y. Chuang, "Deepco3: Deep instance co-segmentation by co-peak search and co-saliency detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[45] B. Li, Z. Sun, Q. Li, Y. Wu, and A. Hu, "Group-wise deep object co-segmentation with co-attention recurrent neural network," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[46] C. Zhang, G. Li, G. Lin, Q. Wu, and R. Yao, "Cyclesegnet: Object co-segmentation with cycle refinement and region correspondence," *IEEE Transactions on Image Processing*, vol. 30, pp. 5652–5664, 2021.

[47] H. Blum, "Biological shape and visual science (part i)," *Journal of theoretical Biology*, vol. 38, no. 2, pp. 205–287, 1973.

[48] K. Siddiqi and S. Pizer, *Medial representations: mathematics, algorithms and applications*. Springer Science & Business Media, 2008, vol. 37.

[49] P. Dimitrov, C. Phillips, and K. Siddiqi, "Robust and efficient skeletal graphs," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, vol. 1. IEEE, 2000, pp. 417–423.

[50] R. A. Katz and S. M. Pizer, "Untangling the blum medial axis transform." *International Journal of Computer Vision*, vol. 55, 2003.

[51] X. Bai, L. J. Latecki, and W.-Y. Liu, "Skeleton pruning by contour partitioning with discrete curve evolution," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 3, pp. 449–462, 2007.

[52] X. Bai and L. J. Latecki, "Discrete skeleton evolution," in *EMMCVPR*, 2007, pp. 362–374.

[53] A. D. Ward and G. Hamarneh, "The groupwise medial axis transform for fuzzy skeletonization and pruning," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 32, no. 6, pp. 1084–1096, 2009.

[54] C. Arcelli and G. S. Di Baja, "A width-independent fast thinning algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 4, pp. 463–474, 1985.

[55] C. Pudney, "Distance-ordered homotopic thinning: a skeletonization algorithm for 3d digital images," *Computer vision and image understanding*, vol. 72, no. 3, pp. 404–413, 1998.

[56] P. Golland, W. Eric, and L. Grimson, "Fixed topology skeletons," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, vol. 1. IEEE, 2000, pp. 10–17.

[57] Y. Y. Tang and X. You, "Skeletonization of ribbon-like shapes based on a new wavelet function," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 25, no. 9, pp. 1118–1133, 2003.

[58] J. W. Brandt and V. R. Algazi, "Continuous skeleton computation by voronoi diagram," *CVGIP: Image understanding*, vol. 55, no. 3, pp. 329–338, 1992.

[59] N. Mayya and V. Rajan, "Voronoi diagrams of polygons: A framework for shape representation," *Journal of Mathematical Imaging and Vision*, vol. 6, pp. 355–378, 1996.

[60] R. L. Ogniewicz and O. Kübler, "Hierarchic voronoi skeletons," *Pattern recognition*, vol. 28, no. 3, pp. 343–359, 1995.

[61] C. Arcelli and G. S. di Baja, "Euclidean skeleton via centre-of-maximal-disc extraction," *Image and Vision Computing*, vol. 11, no. 3, pp. 163–173, 1993.

[62] G. Malandain and S. Fernández-Vidal, "Euclidean skeletons," *Image and vision computing*, vol. 16, no. 5, pp. 317–327, 1998.

[63] W.-P. Choi, K.-M. Lam, and W.-C. Siu, "Extraction of the euclidean skeleton based on a connectivity criterion," *Pattern Recognition*, vol. 36, no. 3, pp. 721–729, 2003.

[64] Y. Ge and J. M. Fitzpatrick, "On the generation of skeletons from discrete euclidean distance maps," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 11, pp. 1055–1066, 1996.

[65] F. Meyer, "Skeletons and perceptual graphs," *Signal Processing*, vol. 16, no. 4, pp. 335–363, 1989.

[66] P. Maragos and R. Schafer, "Morphological skeleton representation and coding of binary images," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 5, pp. 1228–1244, 1986.

[67] J. Goutsias and D. Schonfeld, "Morphological representation of discrete and binary images," *IEEE Transactions on Signal Processing*, vol. 39, no. 6, pp. 1369–1379, 1991.

[68] R. Kresch and D. Malah, "Morphological reduction of skeleton redundancy," *Signal Processing*, vol. 38, no. 1, pp. 143–151, 1994.

[69] W. Shen, X. Bai, R. Hu, H. Wang, and L. J. Latecki, "Skeleton growing and pruning with bending potential ratio," *Pattern Recognition*, vol. 44, no. 2, pp. 196–209, 2011.

[70] S. Tsogkas and I. Kokkinos, "Learning-based symmetry detection in natural images," in *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VII 12*. Springer, 2012, pp. 41–54.

[71] W. Shen, X. Bai, X. Yang, and L. J. Latecki, "Skeleton pruning as trade-off between skeleton simplicity and reconstruction error," *Science China Information Sciences*, vol. 56, pp. 1–14, 2013.

[72] W. Shen, K. Zhao, Y. Jiang, Y. Wang, Z. Zhang, and X. Bai, "Object skeleton extraction in natural images by fusing scale-associated deep side outputs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 222–230.

[73] K. Zhao, W. Shen, S. Gao, D. Li, and M.-M. Cheng, "Hi-fi: Hierarchical feature integration for skeleton detection," *arXiv preprint arXiv:1801.01849*, 2018.

[74] N. H. Nguyen, "U-net based skeletonization and bag of tricks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2105–2109.

[75] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman, "Sift flow: Dense correspondence across different scenes," in *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part III 10*. Springer, 2008, pp. 28–42.

[76] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[77] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, pp. 91–110, 2004.

[78] W.-Y. D. Lin, M.-M. Cheng, J. Lu, H. Yang, M. N. Do, and P. Torr, "Bilateral functions for global motion modeling," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13*. Springer, 2014, pp. 341–356.

[79] D. Mishkin, J. Matas, and M. Perdoch, "Mods: Fast and robust method for two-view matching," *Computer vision and image understanding*, vol. 141, pp. 81–93, 2015.

[80] J.-M. Morel and G. Yu, "Asift: A new framework for fully affine invariant image comparison," *SIAM journal on imaging sciences*, vol. 2, no. 2, pp. 438–469, 2009.

[81] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2758–2766.

[82] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2462–2470.

[83] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8934–8943.

[84] I. Melekhov, A. Tiulpin, T. Sattler, M. Pollefeys, E. Rahtu, and J. Kannala, "Dgc-net: Dense geometric correspondence network," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2019, pp. 1034–1042.

[85] P. Truong, M. Danelljan, L. V. Gool, and R. Timofte, "Gocor: Bringing globally optimized correspondence volumes into your neural network," *Advances in Neural Information Processing Systems*, vol. 33, pp. 14 278–14 290, 2020.

[86] P. Truong, M. Danelljan, L. Van Gool, and R. Timofte, "Learning accurate dense correspondences and when to trust them," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5714–5724.

[87] P. Truong, M. Danelljan, R. Timofte, and L. Van Gool, "Pdc-net+: Enhanced probabilistic dense correspondence network," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[88] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[89] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[90] A. Siarohin, S. Roy, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, "Motion-supervised co-part segmentation," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 9650–9657.

[91] D. Pathak, E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional multi-class multiple instance learning," *arXiv preprint arXiv:1412.7144*, 2014.

[92] X. Chen, A. Shrivastava, and A. Gupta, "Enriching visual knowledge bases via object discovery and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 2027–2034.

[93] J. Dai, K. He, and J. Sun, "Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1635–1643.

[94] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, "What's the point: Semantic segmentation with point supervision," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*. Springer, 2016, pp. 549–565.

[95] G. Papandreou, L.-C. Chen, K. P. Murphy, and A. L. Yuille, "Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1742–1750.

[96] C. Rother, V. Kolmogorov, and A. Blake, "" grabcut" interactive foreground extraction using iterated graph cuts," *ACM transactions on graphics (TOG)*, vol. 23, no. 3, pp. 309–314, 2004.

[97] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, vol. 1. IEEE, 2001, pp. 105–112.

[98] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "icoseg: Interactive co-segmentation with intelligent scribble guidance," in *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2010, pp. 3169–3176.

[99] D. Lin, J. Dai, J. Jia, K. He, and J. Sun, "Scribblesup: Scribble-supervised convolutional networks for semantic segmentation," in *Proceedings of the*

IEEE conference on computer vision and pattern recognition, 2016, pp. 3159–3167.

[100] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[101] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels," Tech. Rep., 2010.

[102] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.

[103] "Pascal-part dataset," http://roozbehm.info/pascal-parts/pascal-parts.html.

[104] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," 2011.

[105] Y. Zhang, Y. Guo, Y. Jin, Y. Luo, Z. He, and H. Lee, "Unsupervised discovery of object landmarks as structural representations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2694–2703.

[106] R. Hu, M. Rohrbach, and T. Darrell, "Segmentation from natural language expressions," in *European Conference on Computer Vision*. Springer, 2016, pp. 108–124.

[107] R. Liu, C. Liu, Y. Bai, and A. L. Yuille, "Clevr-ref+: Diagnosing visual reasoning with referring expressions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4185–4194.

[108] L. Yu, Z. Lin, X. Shen, J. Yang, X. Lu, M. Bansal, and T. L. Berg, "Mattnet: Modular attention network for referring expression comprehension," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1307–1315.