MULTIPLE PARENTING IDENTIFICATION IN IMAGE PHYLOGENY

A. Oliveira[1], *P. Ferrara*[2,3], *A. De Rosa*[3], *A. Piva*[2], *M. Barni*[4], *S. Goldenstein*[1], *Z. Dias*[1], *A. Rocha*[1]

[1]Institute of Computing – University of Campinas – Campinas, Brazil
[2]National Inter-University Consortium for Telecommunications – Florence, Italy
[3]Department of Information Engineering – University of Florence – Florence, Italy
[4]Department of Information Engineering and Mathematics – University of Siena – Siena, Italy

ABSTRACT

Image phylogeny deals with tracing back parent-child relationships among near duplicates, images that share the same semantic content. This approach results in a visual structure showing the inheritance of semantic content among images, called phylogeny tree. In this paper, we extend upon the image phylogeny's original formulation, which considers that an image may inherit content from only a single parent, to deal with situations whereby an image may inherit it from multiple different parents. Our objective is to find the multiple parenting relationships in a set of images, a problem which we refer to as multiple parenting phylogeny. The proposed solution works by first identifying near-duplicate groups and reconstructing their phylogenies; then among the found groups we determine the one(s) representing the composition images; finally, we detect the parenting relations between those compositions and the source images used to create them.

Index Terms— Image Phylogeny, Near Duplicate Detection, Forgery Detection

1. INTRODUCTION

The near duplicates of a document are defined as all the versions of the document with the same semantic content but differing by small transformations. Given that images carry on a great deal of semantic information and are very easy to obtain and share, the problem of near duplicate detection is receiving increasing attention. Beyond the detection of near duplicates, some research groups [1, 2, 3] modelled how the images in a set of near duplicates generated each other, identifying in this process the original image, which is the one that spawned the whole set. Dias et al. approached this problem [3, 4] from a point of view of *phylogeny*: just as organisms evolve in Biology, a document can change over time to slightly different versions of itself, where each of these versions can generate other versions. Hence, the aim of phylogeny is to reconstruct the Image Phylogeny Tree of a set of near duplicate documents.

The original image phylogeny model assumes that one image can only inherit content from one single image. Because of this assumption, each node in the phylogeny structure has at most one parent, making it a tree. This assumption, although acceptable in most cases, is not always valid. Content from multiple *source* images can be combined, for example in montages or splicing, to create what is called an *image composition*. We name the reconstruction of the graph in presence of images coming from multiple sources as *Multiple Parenting Phylogeny*, in order to highlight the main difference with the original Image Phylogeny approach.

Discovering multiple parenting relationships also has many applications in practical scenarios, such as content tracking, forensics or copyright enforcement. As an example, we may consider pornographic compositions using personalities (such as celebrities or politicians), with the purpose of public shaming. By taking advantage of the large amount of images shared by users and using multiple parenting phylogeny, it is possible not only to identify the image as a composition, but also to retrieve the source images used to create it. Knowing such sources serves as hard evidence that the pornographic image is a forgery, clearing the name of the victim. Finally, multiple parenting phylogeny can be easily extended to other types of media as well, such as texts, audio or videos, providing applications in those domains, such as plagiarism detection.

Multiple parenting phylogeny is a natural extension of the image phylogeny problem, allowing us to find the relationships not only between images with essentially the same content (near duplicates) but also those of seemingly unrelated content. This raises many challenges not present in image phylogeny, since we need to find relationships among images with no prior information about the amount of content they share. To do this, it is necessary to accurately reconstruct the phylogenies existent in a set of images, as well as precisely localize and compare the shared content between images. By overcoming those challenges, we can go even further in the analysis of the evolution of documents on the internet, and specially, how a new content is created by the combination of existing sources.

To find the multiple parenting relationships in a set of images, we introduce a method that works by grouping images into well-separated sets of near-duplicates, reconstructing their phylogenies, and pointing out which groups are compositions, finding the sources used to create them.

2. METHOD

The main objective of multiple parenting phylogeny is to discover the inheritance of content between compositions and their sources. In the most common scenario, we have three types of images: hosts and aliens (both the *source* images), and compositions, each one related to a near-duplicate set. The composition is the result of inserting a portion of an alien image into a host image. We can divide this particular scenario in three major problems: (1) determining

---

We thank the support of CAPES, CNPq (#307018/2010-5, #304352/2012-8, #306730/2012-0, #477692/2012-5, #477662/2013-7, #483370/2013-4), FAPESP (#2010/05647-4, #2013/08293-7 and #2013/21251-1), Microsoft Research and the European Union through the REWIND (REVerse engineering of audio-VIsual coNtent Data) project. The project REWIND acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number: 268478.

978-1-4799-5751-4/14/$31.00 ©2014 IEEE

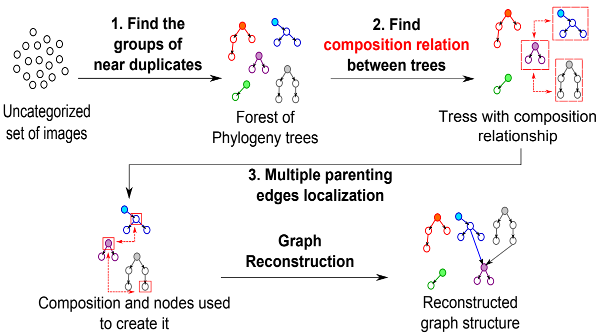

Fig. 1. Pipeline of our multiple parenting approach. Step 1 separates the images of the set in groups of similar semantic content, using a phylogeny forest algorithm. Step 2 looks for shared content between the scenes of each tree, classifying the trees in compositions and sources. Step 3 searches for the nodes used to generate the composition. Finally, the graph with the multiple parenting relationships is reconstructed.

the images depicting the same semantic content; (2) analyzing each group of near duplicates and pointing out whether the images therein are sources (images that could be used in a composition), compositions or unrelated to the rest; and finally, (3) inspecting each classified group and identifying the phylogeny relationship of the images therein and the ones actually used to create the compositions.

To solve the aforementioned problems, in this section, we introduce a 3-step method to automatically (1) find and (2) group near-duplicate images; and (3) classify nodes as host images (used as backgrounds in a composition), aliens (image pieces spliced with other images) or compositions (result of a combination of host and alien images). The following sections show details of each step of the method, whose full pipeline is depicted in Figure 1.

2.1. Finding near-duplicate groups

Image phylogeny aims at finding a structure, the phylogeny tree, to describe a near-duplicate set, with the original image that generated the set as the root. To build the phylogeny tree, Dias et al. proposed [3, 4] a 2-step method: calculation of a dissimilarity matrix for all analyzed images and reconstruction of the tree.

Given two images, the dissimilarity d between them is a measure describing their parenting relationship. A low value of d between an ordered pair of images means that it is likely that one is the parent of the other in the tree. Let there be a family T of image transformations, with $T_{\vec{\beta}}$ a subset of transformations belonging to the family T, parameterized by the values of $\vec{\beta}$. The dissimilarity between images I_A and I_B is defined as

$$d(I_A, I_B) = \min_{T_{\vec{\beta}}} \left| I_B - T_{\vec{\beta}}(I_A) \right|_{\text{pointwise comparison } L}, \qquad (1)$$

for all possible values $\vec{\beta}$ of parameters of T. To find $d(I_A, I_B)$, it is first necessary to find the parameters $\vec{\beta}$ that minimize the difference between $T_{\vec{\beta}}(I_A)$ and I_B. The family of transformations considered by Dias et al. [4] comprised: resampling, cropping and affine transformation, contrast, brightness, gamma correction, and compression.

To map I_A onto I_B's domain for estimating resampling, cropping and affine transformation, a registering approach based on RANSAC [5], with keypoints detected through algorithms such as SIFT [6] or SURF [7] is used. Color transformations and compression parameters are also estimated. After these operations, $T_{\vec{\beta}}(I_A)$ is obtained and compared with I_B using a pointwise image comparison metric L. Dias et al. [3, 4] used as comparison metric L the *Minimum Squared Error*, but other metrics, such as the Euclidean Distance, can also be used.

With a full dissimilarity matrix for every pair of images in the set, the authors proposed an extended Kruskal minimum spanning tree algorithm to work on directed graphs and find the image phylogeny tree associated with the set. Such solution considers that in a set of images with the same content, all of them are near duplicates.

The problem of phylogeny forests arises when, in a set of images with the same semantic content, not all of them are associated by the same acquisition process (come from one original source). This happens when multiple pictures are taken from the same scene, with the same camera and different parameters or from the same scene with different cameras. In this case, the set of images will have multiple phylogeny trees, and a forest algorithm is responsible for identifying them. Dias et al. proposed [8] an approach based on the modification of their oriented Kruskal algorithm. This modification works by adding edges to a tree only if the weight of that edge is not higher than an adaptive threshold calculated on the edge weights already added to the solution. This threshold is dependent on the higher dissimilarity between images that have related content but are from different sources. Therefore, it is intuitive to see how a forest algorithm would also work in a scenario of different trees of unrelated semantic content.

2.2. Group classification

After finding groups of near duplicates we still have no information about the relations between compositions and parent images. Also, the dissimilarity measure is unsuitable to discover those relations because it is strongly dependent on the type of composition (e.g., the size of the tampering region). A content-dependent descriptor, possibly invariant to geometrical, color and compression transformation, is needed to detect shared content among groups of images. To detect composition trees, we have adopted a SIFT-based approach [9].

For simplicity, we assume that a composition image is obtained by the composition of only two images, by copying a patch or portion cut from an image (*alien*) to another one (*host*). Moreover it is reasonable to assume that the patch is small with respect to the background, belonging to the host image.

Ideally, since each root of the tree obtained at the previous step summarizes the content of each tree, we perform a pairwise comparison between all possible combinations of the roots. The comparison is based on the extraction of keypoints and their SIFT descriptors, the matching of the keypoints as proposed by Lowe [6], and their clustering. In practice, however, this strategy proves to be error-prone and not sufficient: the presence of outliers coming from the matching strategy requires a robust clustering. Therefore, we adopt a J-Linkage clustering algorithm [10] to cluster keypoints in function of the estimated geometrical transformation applied to parent images. The method consists of generating a fixed number of geometrical transformation hypothesis by a random sampling of a neighborhood of matched keypoints. After that, for each pair of matched keypoints, a preference set vector (PS) is defined indicating which transformations the pair prefers. The PSs are used in a hierarchical agglomerative clustering to estimate the transformations. This algorithm starts by assigning each PS to a cluster; then, for each step of the algorithm, the two clusters with smallest distance are merged. The PS of a cluster is computed as the intersection of the preference sets of matched pairs, and the distance between two clusters is computed as the *Jaccard* distance between the respective preference sets.

J-Linkage presents some advantages: it is robust to the presence of outliers, it can be easily applied in case of more than two parent images and it does not need a priori information about the percentage of outliers, as in the case of RANSAC. This last property is suitable in our scenario because the number of outliers changes in function of the matched keypoints (i.e., on the content). For instance, when comparing images whose content is completely unrelated, all matched keypoints are outliers; conversely, when comparing a composition image with its parent image, the vast majority of the matched keypoints are inliers, rather than outliers. The main J-Linkage's limitation is the generation of small clusters of keypoints with degenerative models, due to the outliers. To reduce their impact, a threshold on the minimum number ($N_c = 5$) of keypoints satisfying the estimated transformation is applied.

Sometimes the aforementioned strategy fails because the composition is obtained from near duplicate of the roots (instead of the roots themselves), which have undergone a set of color, compression and geometrical transformations, altering the SIFT descriptors and making some matching undetectable. Therefore, we also extend the useful information from the roots to the trees, by randomly sampling an image (node) from each tree. The test based on SIFTs is repeated a fixed number of times, and all those trees which have at least one image with more than one relation with other images is classified as composition tree.

Finally, to classify the parent trees as either alien or host, we employ the dissimilarities $d(I_A, I_B)$, evaluated at the previous step for the pairs composition-host and composition-alien roots. Since patches are small with respect to the background, the dissimilarity between host and composition root is lower than the dissimilarity between alien and composition ones.

2.3. Parents identification

In the previous step, relations between different groups of images are established, but they miss information about the exact sources that have generated the composition. Due to the different nature of the relations between host and composition groups and between alien and composition ones, we employ both dissimilarity measure and SIFT matching based approaches to trace the images which have exactly generated a composition. In the case of the host parent, we employ the dissimilarity $d(I_A, I_B)$ rather than a SIFT-based approach (which introduces a considerable computational effort), by observing that the host parent of the composition is the node of the host tree that has the lowest $d(I_A, I_B)$ with the composition root. This constraint is acceptable if we assume that the content coming from the alien parent is relatively small with respect to the background.

In case of the alien parent, we need to localize the shared content inside the composition root and evaluate the dissimilarity $d(I_A, I_B)$ only on that portion of content, to avoid noise due to the background belonging to the host. We use the same SIFT-based approach as in the previous step, by comparing all alien images with the related composition root. After identifying the cluster of matches between one of the aliens and the composition, we use the mean of the distances between the matches in the cluster as the dissimilarity between the shared content of both images. We select as the alien parent the one with the smallest dissimilarity among the tested nodes.

3. EXPERIMENTAL SETUP

This section presents the validation protocol for all experiments.

3.1. Dataset and Test Cases

The dataset[1] used comprises 100 host and 150 alien base images, as well as 5000 compositions. The host images are outdoor and indoor background scenes, such as rooms, streets or fields, obtained from the *Inria Holidays* [11] dataset. The aliens are images of varied objects, such as people, cars or animals, in common backgrounds. Those images were collected from *Berkley Segmentation* [12] and *Graz-02* [13] datasets, with segmentation masks from *Interactive Segmentation Tool* [14] and *Inria Annotations* [15], respectively.

The dataset also has a number of phylogeny tree files, associated with the base images, which describe a tree topology and the parameters of a set of image processing operations. Using the base image as root, the images are transformed following the topology and the operations described in order to generate the whole phylogeny tree. The operations and their parameter ranges were the same used in Dias et al. work [4]. All host and alien base images have 25 phylogeny trees (5 different topologies, with 5 parameter variations each) of 25 nodes. Because compositions are unique, they only have a single phylogeny tree of 25 nodes.

The test cases are phylogeny forests of 75 nodes, consisting of a host, an alien, and a composition tree. To generate one, we first randomly select a pair of host and alien base images, as well as two of their phylogeny tree files, and build the respective trees. Two random host and alien nodes are then picked from each tree to create the composition, by automatically cutting the object from the alien parent (using its segmentation mask) and pasting it randomly in the host parent. Composition types differ by pasting method, being either *direct pasting*, where the object is cut and pasted into the host with no changes whatsoever, or *poisson blending*, where the pasted object is blended into the host using Pérez et al.'s [16] method of gradient adjustment. Finally, the composition phylogeny tree is built, completing the generation of the test case. In this work, 300 direct pasting and 300 poisson blending test cases were used.

3.2. Metrics

To evaluate the accuracy of the groups and the reconstructed phylogeny forest, we use the metrics *roots*, *edges*, *leaves* and *ancestry* defined by Dias et al. [3, 4] and a new *subset* metric, all measured considering we have the *groundtruth* forest. The *roots* metric checks whether the roots in the reconstructed and in the groundtruth forests are the same, while the *leaves* metric does the same for the leaves. The *edges* metric measures the percentage of right parenting relationships found in the reconstructed forest. Finally, the *ancestry* metric evaluates if each node in the reconstructed forest has the same set of ancestors as in the groundtruth. The *subset* measure, developed for this work, measures if images with the same semantic content end up in the same trees in the reconstructed forest, i.e., it measures if the image phylogeny forest algorithm correctly separates the images in meaningful groups. First, we define the set:

$$\begin{aligned}\sigma(F_R, F_{GT}) = \{(I_A, I_B) | \pi(I_A, F_R) = I_B \quad \wedge \\ \tau(I_A, F_{GT}) = \tau(I_B, F_{GT}), \\ \forall I_A \in F_R \setminus \rho(F_R)\}\end{aligned} \tag{2}$$

where F_x, with $x \in \{R, GT\}$, is a reconstructed (R) or a ground truth (GT) forest, (I_A, I_B) is a generic couple of images and $\pi(I, F)$ is a function returning the parent of an image I in the forest F. Finally, $\tau(I, F)$ returns the tree to which the image I belongs in F,

---

[1] The dataset and test cases used in this work are available at http://dx.doi.org/10.6084/m9.figshare.1050094

and $\rho(F)$ gives the roots of F. The $subset$ metric is defined as:

$$subset = \frac{|\sigma(F_R, F_G T)|}{|F_R \setminus \rho(F_R)|} \tag{3}$$

The subset metric is important because it gives information about the *separation* of the host, alien and composition subset.

To evaluate the results of our multiple parenting approach we introduce the metrics *composition root* (CR), *host parent* (HP) and *alien parent* (AP), which test if such nodes were correctly found in each test case. Additionally, we employ the metrics *composition node* (CN), *host node* (HN) and *alien node* (AN), used to check if the composition root and host and alien parents are, respectively, composition, host and alien images. This second set of metrics is used to evaluate the classification of the trees.

4. RESULTS AND DISCUSSION

This section shows the experiments and results for multiple parenting identification. First, we present results for finding the groups and the phylogeny relations within each group. Then we show results for multiple parenting identification, rates with which the proposed method correctly classifies the trees in host/alien/composition, and its accuracy at identifying the nodes that generated compositions.

4.1. Forest Algorithm Results

Since there is no solution in the literature yet for the multiple parenting phylogeny problem, we consider two different forest algorithms in the experiments. The first one is a modification of the Oriented Kruskal [3, 4] to extract from a dissimilarity matrix exactly three trees, under the assumption that we know the number of trees in the forest, which we call K3T. The other is the automatic oriented Kruskal (AOK) as presented by Dias et al. [8], which tries to automatically identify the number of trees in the forest. As previously discussed, the AOK algorithm relies on a threshold parameter for adding new edges to the phylogeny forest. Using a smaller and completely separated set of 100 test cases, it was found that the best value for this parameter was 3.0. K3T is used just as an upper bound as in practice everything needs to be automatically calculated and we do not know the number of trees in the forest. It was also observed that, in most cases, the number of trees found by AOK was equal to or very close to three indicating that, even though AOK is automatic, it still has good results in finding the correct number of trees in the forest, making it a safe choice as the image grouping algorithm. Table 1 shows results for the reconstructed phylogeny forests, divided by direct pasting and poisson blending types of image composition.

Table 1. Forest algorithm results for finding near-duplicate groups.

		Metrics				
Type	**Algorithm**	**root**	**edges**	**leaves**	**ancestry**	**subset**
Direct	AOK	81.6%	74.3%	81.4%	65.5%	99.9%
	K3T	83.9%	74.4%	81.4%	66.6%	99.9%
Poisson	AOK	78.7%	74.5%	81.3%	63.2%	99.7%
	K3T	82.2%	74.6%	81.4%	66.1%	99.9%

Both algorithms show similar and good results, with K3T slightly better in the *roots* and *ancestry* metrics as expected. It is also important to note that K3T and AOK present nearly perfect results for the subset metrics which means that both algorithms are effective for separating the host, alien and composition trees. This is specially important for tree identification, as a bad separation of trees could lead to a wrong classification further on in the method.

4.2. Multiple Parenting Results

Table 2 shows the results for tree classification and multiple parenting identification. As detailed in Section 2.2, we classify the trees found by the forest algorithm applied in the first step by choosing random nodes of each tree and comparing their content to find shared objects, repeating this process a fixed number of times. The algorithm was tested with the number of repetitions: $\{1, 3, 5, 10, 15, 20, 25\}$. As there were no obvious gains of accuracy with more repetitions, it was decided to fix the number in five, as the computational cost tends to rise as more repetitions are used.

Table 2. Multiple parenting results.

		Metrics					
Type	**Algorithm**	**CR**	**CN**	**HP**	**HN**	**AP**	**AN**
Direct	**AOK**	73.0%	91.7%	76.0%	93.0%	33.7%	98.3%
	K3T	74.7%	92.7%	78.0%	94.3%	34.3%	99.0%
Poisson	**AOK**	66.3%	85.3%	73.0%	88.3%	11.3%	98.7%
	K3T	66.3%	87.0%	75.7%	88.3%	11.3%	99.3%

The algorithms present similar performance for the two types of compositions. Considering that in about 30% of the test cases AOK does not find three trees, those results are important to show that even when the number of trees found is incorrect, the classification of the trees, as shown by the *CN*, *HN* and *AN* metrics, still presents good accuracy. This is due to the robust process of classification that counts the number of content relationships between different trees in the forest, which keeps valid even if a tree is broken into sub-trees. When the composition tree is split into two trees, the low dissimilarity between the two might lead to wrongly classifying one of them as host tree. However, by comparing AOK with K3T results, those cases have small impact on the overall accuracy of the method.

We have good results in finding the original composition and its host parent, as shown by the CR and HP results. The CR value, in special, is dependent on the roots found by the forest algorithm, as we always choose the root of the tree identified as composition as the original composition. Finally, even though the proposed method shows very good results in identifying the alien tree, as in about 99% of the cases the alien parent identified is one of the alien nodes, we still are not very good at finding the correct alien node used in the composition process. As discussed before, we currently use the SIFT distance of the shared content existent between composition and alien as the comparison metric. This measure is not perfect at identifying the transformations the shared content went through, which might lead to a wrong classification.

5. CONCLUSION

In this work, we presented a novel method for the identification of multiple parenting relationships in sets of images. It combines a phylogeny forest approach for group separation with object detection techniques for identification of shared content between images. Using this pipeline, the final result is a graph structure showing both the relationships between images with the same semantic and images with partially shared content.

The proposed method shows promising results in finding the different semantic groups (with an effectiveness exceeding 99%) and discovering the relationships between those groups (at least 85% of the cases), labeling them as compositions, hosts and aliens. Our future efforts will focus on finding the correct alien parent with a higher accuracy, by improving the estimation of the shared content region as well as our metrics to compare them, and expanding the proposed method to work with other types of compositions.

6. REFERENCES

[1] L. Kennedy and S.-F. Chang, "Internet image archaeology: Automatically tracing the manipulation history of photographs on the web," in *ACM International Conference on Multimedia*, 2008, pp. 349–358.

[2] A. De Rosa, F. Uccheddu, A. Costanzo, A. Piva, and M. Barni, "Exploring image dependencies: a new challenge in image forensics," in *SPIE Media Forensics and Security II*, 2010, p. 75410.

[3] Z. Dias, A. Rocha, and S. Goldenstein, "First steps toward image phylogeny," in *IEEE International Workshop on Information Forensics and Security*, 2010, pp. 1–6.

[4] Z. Dias, A. Rocha, and S. Goldenstein, "Image phylogeny by minimal spanning trees," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 774–788, 2012.

[5] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Springer International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[7] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Elsevier Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[8] Z. Dias, A. Rocha, and S. Goldenstein, "Toward image phylogeny forests: Automatically recovering semantically similar image relationships," *Elsevier Forensic Science International*, vol. 231, no. 13, pp. 178–189, 2013.

[9] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, L. Del Tongo, and G. Serra, "Copy-move forgery detection and localization by means of robust clustering with J-Linkage," *Elsevier Signal Processing: Image Communication*, vol. 28, no. 6, pp. 659–669, 2013.

[10] R. Toldo and A. Fusiello, "Robust multiple structures estimation with J-Linkage," in *Springer European Conference on Computer Vision*, 2008, pp. 537–547.

[11] H. Jgou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Springer European Conference on Computer Vision*, 2008, pp. 304–317.

[12] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *IEEE International Conference on Computer Vision*, 2001, vol. 2, pp. 416–423.

[13] A. Opelt, A. Pinz, M. Fussenegger, and P. Auer, "Generic object recognition with boosting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 3, pp. 416–431, 2006.

[14] K. McGuinness and N. E. O'Connor, "A comparative evaluation of interactive segmentation algorithms," *Elsevier Pattern Recognition*, vol. 43, no. 2, pp. 434–444, 2010.

[15] M. Marszatek and C. Schmid, "Accurate object localization with shape masks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[16] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," in *ACM Special Interest Group on GRAPHics and Interactive Techniques*, 2003, pp. 313–318.