Designing from Motivation: Exploring Large Scale Tagged Data Collection through Social Monetization Computing

by

Qiong Wu

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Computing Science

University of Alberta

Abstract

With the exponential growth of web image data, image tagging is becoming crucial in many image based applications such as object recognition and content-based image retrieval. However, despite the great progress achieved in automatic recognition technologies, none has yet provided a satisfactory solution to be widely useful in solving generic image recognition problems. Automatic technologies usually make certain assumptions, such as a limited number of object categories and how many objects there are in an image. With the goal of tagging generic images, so far, only manual tagging can provide precise image descriptions. However, the cost and tediousness of manual tagging is the major concern.

The first effort to motivate people to tag images is the ESP game, proposed by Luis von Ahun. In the same vein, we ask the same question how can we motivate people to tag web images, which belongs to the research field of collective intelligence. So far, crowdsourcing, human computation (ESP game) and social computing are three major methods resolving the problem of motivating people to work collaboratively and to produce something intelligent. However, none of them can achieve the goal of collecting large scale tagged images at high quality for low cost.

In this thesis, we propose a Social Monetization Computing (SMC) model, which incorporates monetary incentives into social computing to guarantee high quality work from both crowdsourcing workers and social web users for a low cost. In addition, we summarize a design guidance of a SMC system. In the light of SMC system design guidelines, we describe the evolutionary design and implementation of an image tagging system, called EyeDentifyIt, driven by image-click-ads framework. A series of usability studies are presented to demonstrate how EyeDentifyIt provides better user motivations, produces higher quality data, and requires less workload from workers, compared to state-of-the-art approaches.

To further reduce workload involved in the image tagging process, we develop an efficient method for automatically parsing fashion images, which resolves three common problems including occlusions, background spills and over smoothing of infrequent labels, in existing fashion parsing methods. The experiment results demonstrate that the proposed method outperforms state-of-the-art clothing parsing methods from both quantity and quality perspectives.

To the people,

for teaching me everything I need to learn.

To the Internet,

for offering me everything I want to know.

To my future Mr.Right,

for allowing me to explore the world on my own.

Acknowledgements

There are many people I would like to thank for making this thesis possible. They gave me guidance, support, encouragement, and help, not only for the days of my Ph.D. life, but also accompanying me for my future life.

First of all, this work would not have been possible without my PhD advisor Pierre Boulanger for his open mind, unconditional support, advice and most importantly humour throughout my studies. I also would like to thank my thesis committee members, Irene Cheng, Abram Hindle, Anup Basu for their advice, comments, and thoroughly going over my thesis in challenging me, pointing me in the right direction and making this work better. Additionally, I would like to thank Xuedong Yang, who is my adviser during my undergraduate studies. Thank him for giving me the opportunity of study abroad in the early days of my undergraduate life.

I am thankful to my collaborators as well as friends Dan Han, Rui Gao, Xida Chen, and JiangWei Yu, who had shared with me their knowledge and passion for inspiring this work, devoted their precious spare time besides their own work and project. I would like to especially thank Rui Gao and Dan Han, who always support and accompany me, and had shared my joy and grief, up and down during this time of my life. In addition, I would like to thank all my friends who I am luck to know during my graduate studies, Maryia Kazakevich, Robyn Taylor, Matthew Hamilton, AmirAli Sharifi, Rui Shen, Ying Xu, Cheng Lei, Ailin Zhou, Bing Xu, Bang Liu, and Idanis Diaz.

Finally, I wish to take this opportunity to express my deepest appreciation to my beloved family. This thesis cannot be done without your endless love, support, and understanding. To them, I dedicate this thesis.

Contents

List of Figures

Chapter 1

Introduction

Tagging images on the web represents a major technological challenge. Every day millions of images are generated and shared on social networks such as Flickr[1], Pinterest[2], and Instagram[3], but current computer vision technologies has not yet provided a satisfactory solution in an automated way to answer the questions “what is that in the image?”, “where can I get it?”, asked by a web user when (s)he proactively queries a particular content of a web image. Accurate description and linking to related information for in-image content is required in many applications, such as online shopping and image search. These limitations come from the inability to correctly tag images automatically or semi-automatically. Image tagging has been researched for several decades, and its solution still remains elusive. Previous efforts on resolving these problems usually make certain assumptions. For example, the contents of images are related to the text appearing in a web page, or they are limited to a number of image category to recognize.

Most recently, deep learning algorithms based on a modified version of neural nets have been more successful at solving some of these problems. With a reported recognition accuracy of 83% in the ILSVRC-2012 (Large Scale Visual Recognition Challenge) challenge [8], deep learning algorithms [1] have shown to be very promising. More

[1] www.flickr.com

[2] www.pinterest.com

[3] www.instagram.com

recently, a recognition rate of 93.3% was demonstrated by researchers at Google in the ILSVRC-2014 challenge [9] using a modified version of deep learning technique.

Regardless of these significant improvements, automatic methods are still based on certain assumptions. All reported performances are based on the benchmark dataset of up to 1000 object categories [10] (small vocabulary size) with images at low level of scene complexity (most images contain one main object). In the ILSVRC-2014 data set [10], each image has 1.6 target objects on average with 0.47 neighbours (adjacent objects of the same category). Each object occupies 35.8% of pixels compared to 24.1% in the PASCAL data set [11], which may be one of the reasons why Google was able to obtaining a higher classification rate than before. The reported best performance for more challenging tasks such as single-object localization in a 1000 classes data set was 74.7% and for multiple object localization in a 200 classes data set was 37.2% [11]. The recognition accuracy drops significantly when images are cluttered, such as the last two images in Figure 1.1. In addition, because images of ILSVRC-2014 are organized using WordNet [12] semantic structure, the recognition is limited to semantic words in WordNet. Mapping from low-level image features to a general semantic world remains unresolved, which is the well-known *"semantic gap"* problem. Studies [13] have reported that discriminating between thousands of image categories is in fact more difficult that discriminating between hundreds. Overall, existing automatic computer vision technologies have not yet provided a satisfactory solution for tagging images in general cases.

As the increases of data in big data era, tagging images at large scale and high quality is becoming more and more crucial for many image based applications, e.g. training computer vision algorithms. Taking unprecedented acceleration of scientific progress in deep learning [1] as an example, it is a result of both advances in computational power and an enormous data gathering effort, such as ImageNet project [14]. Till April 2010, ImageNet has collected 11 million images for over 15,000 synsets [14]. Existing studies [15] also shows that both quality and quantity of tagged data affect the performance of an intelligent model.

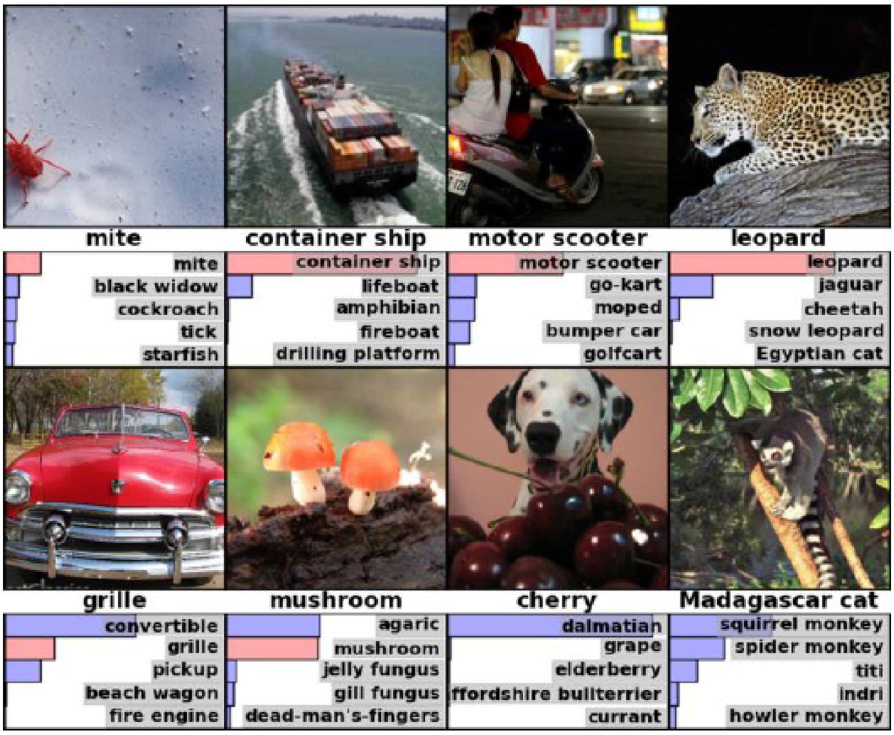


FIGURE 1.1: Examples of ILSVRC-2012 test images and recognized tags by the winning algorithm [1]. The correct tag is given under each image, and the probability assigned to the correct tag is shown with red bar (if it happens to be in the top 5).

Currently the only method that can obtain reliable tagged images is manual techniques, which are tedious and extremely costly. For example, in order to collect $40,000$ image categories with $10,000$ images per category, it takes around 19 years[4] to collect such data, assuming labeling speed is 2 images/sec and it needs three people to label for verification. Such work is tedious, costly and workers have no motivation. Therefore, traditional manual tagging techniques cannot satisfy the need of tagging images at large scale and high quality.

So far, the most common way to collect such data is crowdsourcing. A task assigner can distribute a task to a large group of workers on crowdsourcing platforms using monetary incentives. Crowdsourcing relies on human workers to complete a job for task rewards, which is extremely costly when the collected data is large scale. Suppose there are $1,000,000$ images need to be tagged with the presence/absence of $100,000$ labels, it costs $\$10$ million even in the optimistic setting of perfect workers who tag at a cost of 10 cents per 1,000 tagging tasks [16]. The fact is that ImageNet [14] collects

[4] $40,000 \times 10,000 \times \frac{3}{2} \approx$19 years

such tags by hiring workers from Amazon Mechanical Turk (MTurk) for 10 cents per 5 tags. Regardless of high cost, crowdsourcing can also lead to potential arbitrarily bad results. To obtain one-time rewards, a malicious worker can submit random answers to a task. This can significantly degrade the quality of the collected data. To address above problem, a job is split into many HITs (Human Intelligence Tasks) and each HIT is assigned to multiple workers so that replicated answers are obtained, which increases the cost for many times.

How to design a system that motivates people to tag images at high quality with a low cost have attracted significant research attention [17–20]. One of the first systems capable of motivating people to tag web images was proposed by Luis von Ahun [21]. He invented a way of incentivizing people to tag images using entertaining through a computer game, named the ESP game. In the game, two players are randomly paired and the goal of the game is to guess what their partner is typing for a given image, as shown in Figure 1.2. This work is so inspiring that it creates a new research field, called Human Computation [2]. However, this computing model is limited by such entertaining incentive as well. First, there are only limited number of people interested in playing such games. Second, games are designed to be fun which means players do not want to do tedious work. Such characteristics of games determine that the collected data lacks a certain level of detail, which needs tedious manual input.

Another way of motivating people to work in called social computing. With the rise of social webs, there are billions of humans generating free data in large scale every data that is easily shared, tracked, and searchable. It has enabled the emergence of surprising new forms of collective intelligence. Utilizing such social webs to motivate people to work is called social computing. It is a low cost solution for collecting large scale data at high quality. However, it is hard to control in what direction a crowd works. It is a challenge to find a right way harnessing such human power to tag images useful for a task assigner. The key is to find the right match between what is input online in such social webs and data useful for the task assigner.

To solve the existing problems in crowdsourcing, human computation and social computing, we propose a new computing model targeted at collecting tagged data at large

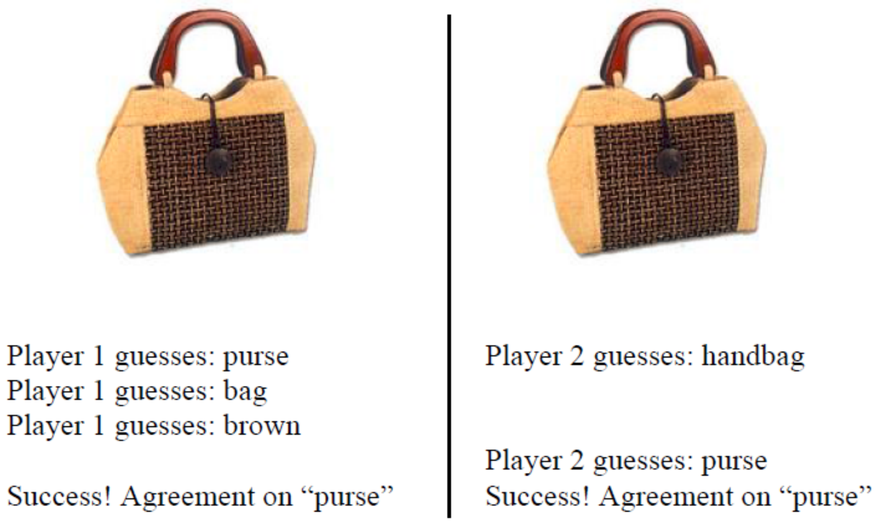


FIGURE 1.2: The process of the ESP game [2]. Two players must type the same string for the given image on the screen (not necessarily at the same time).

scale and high quality. Our work is based on three observations. First, social computing is a low cost solution for large scale data collection, but it is hard to control in what direction a crowd works. Crowdsourcing demonstrates that the monetary incentive is a good way to control the working direction and working quality of workers. In this context, we realized that it could be possible, if designed properly, to harness the working direction and working quality of users in social computing by introducing monetary incentives. Secondly, it is possible that such monetary incentives may not be provided by the task assigner, which can reduce the cost of collecting data. Third, to resolve the tediousness problem, tagging tools should be designed to reduce the workload for workers. In general, the objective of this thesis is to motivate large number of people to tag images at large scale and high quality, for low cost and low workload.

1.1 Objectives

The goal of this thesis is to develop a new methodology that helps a task assigner to collect large scale tagged data on the World Wide Web at high quality with a low cost. In order to do so, in this thesis we explore new ways to perform tasks as below:

- **Tagging data useful for a task assigner**: Data is tagged by workers in a way useful a task assigner;
- **Large scale and high quality**: Collected data is large scale and high quality, which is a common requirement for many applications, such as training machines;
- **Low cost**: Allow a task assigner to collect wanted data with low cost;
- **Low workload**: Allow workers to finish a task with low workload;
- **Open and dynamic**: The collected data can be easily contributed by general web users and can continue to grow in size over time;
- **Motivating participants**: General web users as well as workers are motivated to generate data useful for a task assigner;
- **Harnessing a crowd power**: A task assigner can use the methodology developed in this thesis to control the working direction of social web participants so that they can produce data useful for the task assigner. Data can be collected with a low cost by harnessing mass free human power in social webs.

1.2 Applications

In this thesis, we are mainly motivated by various applications and technologies that are driven by large scale tagged images such as:

- **Training machines:** Large databases of pixel level tagged images are needed for training machine learning algorithms;
- **Content-based image retrieval:** For image search over the web, high-quality tagged images could dramatically increase the accuracy and efficiency of current search engines, resolving semantic gap problem in content-based image retrieval;
- **E-commerce:** The e-commerce opportunities for this system are huge. With hundreds of billions of dollars being spend on commercial products, there are exceptional value for applications that provide the ability to automatically identify and retrieve similar products from images. For example, when a user wishes to find or buy a specific product, (s)he would take a picture of the object and then upload this image on a search engine to find all similar items;

- **Accessibility:** Visually impaired individuals surfing the web need verbal descriptions of images that can be read out loud to understand the contents.

1.3 The Thesis Contributions

To address the problem of tagging images at large scale and high quality, we followed the evolutionary and iterative way of exploring methodology for a system design and development. In this thesis, our main contributions include:

1. **Social Monetization Computing (SMC) model for large scale tagged data collection** More specifically, the contributions of this part mainly include:
   - Proposed a novel SMC model in collective intelligence by introducing social monetization to social computing for collecting data at large scale and high quality needed by a task assigner;
   - Evaluated the model through a series of formative studies, and presented the impact of SMC model on crowdsourcing workers and social web users;
   - Developed design guidelines for a SMC system, according to findings from evaluation results.

2. **EyeDentifyIt – Utilizing Image-Click-Ads Framework for Image Tagging** As a case study, we provided a paradigm for applying SMC model to solve tagging images at large scale and high quality, following the design guideline of SMC system described above. The contributions of this part mainly include:
   - Introduced a new incentive for motivating people to tag images;
   - Integrated a semi-automated segmentation and an automated recognition model with the online system to reduce the vertical workload for the image tagging task;
   - Applied a perceptual hashing algorithm into the online system to reduce the horizontal workload among workers;
   - Demonstrated the prototype provides better user motivations, higher data quality and requires less workload than state-of-the-art crowdsourcing and social computing methods.

3. **A New Method for Automatic Clothing Tagging** To further reduce the workload following the design guidance of a SMC system, we improved the system EyeDentifyIt by a new automatic clothing segmentation method. The proposed method can automatically segment a fashion image into regions and assign tags to corresponding regions. The contributions of this part are as follows:
    - Proposed a deviation of Markov Random Field (MRF), named re-weighted MRF, for parsing clothing images, by introducing 1)two new priors including background prior and occlusion prior to resolve background spill and occlusion problem in clothing parsing; 2)a re-weighted pairwise term in the MRF model to justify infrequent labels in training dataset;
    - Evaluated the proposed method by comparing between different versions of re-weighted MRF and between re-weighted MRF and CRF from both quantitative and qualitative perspectives;
    - Demonstrated that MRF performs better than CRF in conditions that the local knowledge is more trust worthy than the statistical model learned from training dataset;
    - Integrated the proposed clothing parsing method into EyeDentifyIt.

1.4 Thesis Structure

The thesis is organized as following. Chapter 2 presents Social Monetization Computing (SMC) model and a case study for an image tagging system, named EyeDentifyIt 1.0. With the integration of various automatic image tagging tools, Chapter 3 presents the design and implementation details of an evolving prototype, named EyeDentifyIt 2.0. Chapter 4 starts with a review of the related literature on automatic clothing parsing and then introduces our new parsing method and how it is integrated in the final prototype, EyeDentifyIt 3.0. Chapter 6 concludes the thesis and discusses possible future research work and improvements.

1.5 Publications During PHD

- Qiong Wu and Pierre Boulanger. “Social Monetization Computing (SMC) for large scale tagged data collection”, submitted to the SIGCHI conference on human factors in computing systems. ACM, 2016.
- Qiong Wu and Pierre Boulanger. “Efficient Fashion Parsing for Real-time Applications”, submitted to 2015 ACM Transactions on Multimedia (TOMM).
- Qiong Wu and Pierre Boulanger. “An Unified Image Tagging System Driven by a Image-Click-Ads User Interface”, IEEE International Symposium on Multimedia (ISM), 2015.
- Qiong Wu, Rui Gao, Xida Chen, and Pierre Boulanger. “Tagging Driven by Interactive Image Discovery: Tagging-Tracking-Learning” In Multimedia (ISM), 2014 IEEE International Symposium on, pp. 179-186. 2014.
- Pierre Boulanger, Qiong Wu and Maryia Kazakevich. “The Theatre of the Twenty-first Century May Well be Virtual and Online” Crafting Interactive Systems: Learning from Digital Art Practice workshop at CHI 2013.
- Qiong Wu, and Pierre Boulanger. “Real-time Estimation of Missing Markers for Reconstruction of Human Motion.” In Proceedings of the XIII Symposium on Virtual and Augmented Reality (SVR), pp. 161-168, 2011.
- Qiong Wu, Pierre Boulanger, Robyn Taylor, and Maryia Kazakevich. “Trickster at the intersection: exploring virtual theatre performance and interaction.” User in Flux Workshop at CHI 2011.
- Qiong Wu, Pierre Boulanger, Maryia Kazakevich, and Robyn Taylor. “A Real-time Performance System for Virtual Theater” In Proceedings of the 2010 ACM workshop on Surreal media and virtual cloning, pp. 3-8, 2010.
- Qiong Wu, Maryia Kazakevich, Robyn Taylor, and Pierre Boulanger. “Interaction with a Virtual Character through Performance Based Animation” In Proceedings of 10th Symposium on Smart Graphics, pp. 285-288, 2010.

Chapter 2

Social Monetization Computing (SMC) Model

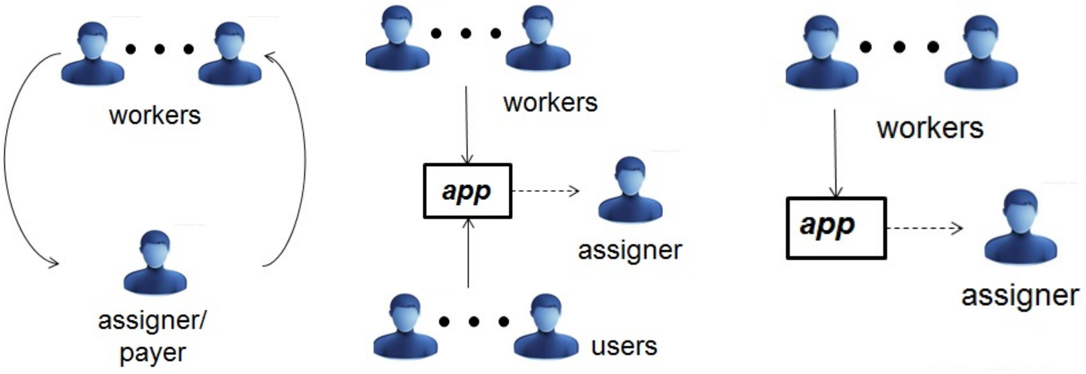
Collecting large scale tagged data is crucial in many fields, such as training machines. Crowdsourcing is increasingly used to finish such tasks. However, it suffers from high cost problem. Social computing is a low cost solution for large scale data collection, but it is hard to control in what direction a crowd works. What applications can best synthesize the strengths of these two approaches, allowing a task assigner to collect tagged data at high quality for low cost? What can happen if we combine crowdsourcing and social computing?

In this chapter, we propose a Social Monetization Computing (SMC) model, which is designed to motivate large number of people to tag data at high quality while reducing cost at the same time. The SMC model transforms the data tagging task into a monetizing tagged data process mediated through a online social communication. To evaluate our model, we implement a practical data tagging system, image tagging. By integrating SMC model into a crowdsourcing image tagging task in an early prototype, the impact of SMC model on crowdsourcing worker's motivation and resulting data quality are studied. According to the feedback of crowdsourcing workers and social web users, design guidelines for a SMC system are summarised.

The contributions of this paper mainly include:

- Proposed a novel SMC model in collective intelligence by introducing social monetization to social computing for resolving data collection needed by a task assigner.
- Provided a paradigm for applying SMC model to solve large scale tagged data collection problem, by presenting a case study for collecting tagged images.
- Presented findings from formative studies, illustrating the impact of SMC model on crowdsourcing workers and social web users.
- According to findings from formative studies, we developed design guidelines for a SMC system.

2.1 Background and Related Work



(A) crowdsourcing (B) social computing (C) human computation

FIGURE 2.1: Comparison of computing model of crowdsourcing, social computing and human computation. Dashed line means implicit computation.

A recent study [22] provided a taxonomy for technologies in collective intelligence by which humans are motivated to collaborate with the aid of computers to accomplish great things, and mainly reviewed this field from three aspects: crowdsourcing, social computing and human computation. We apply the same taxonomy here as tagging large scale data at high quality is one kind of tasks that can only be accomplished by utilizing group collaborations.

2.1.1 Crowdsourcing

A definition of crowdsourcing can be found from Howe's web site[1]:

> "Crowdsourcing is the act of taking a job traditionally performed by a designated agent (usually an employee) and out-sourcing it to an undefined generally large group of people in the form of an open call"

As shown in Figure 2.1a, crowdsourcing solves a computation problem by allowing a task assigner to distribute a task to a large group of workers using monetary incentive. It is a simple explicit computing model as workers only do the task for task rewards. In recent years, it is the most common method for collecting tagged data at large scale. For example, ImageNet [14] collects tens of millions of annotated images by leveraging crowdsourcing platform Amazon Mechanical Turk[2] (MTurk). To guarantee that workers were not making mistakes at following the instructions, they use multiple users independently to tag the same image until there was a convincing majority of the votes for each tagged data. Such voting scheme dramatically increases the cost of data tagging task.

There are growing research interests on resolving various problems existing in crowdsourcing systems, e.g. how to estimate data tagging quality [23–25], how to combine results from multiple workers [26–28], how to reduce cost [29] causing by voting, how to select the next data to be tagged [15, 30], and how to merge machine and human intelligence [31].

However, almost all studies are within the domain of crowdsourcing model itself. In such model, there is always a trade off between monetary costs, sample size, and work quality, and win-win situation for all is hard to achieve. For example, Harris [32] found that increasing financial incentives can improve tagging quality if designed appropriately. Mason and Watts [33] showed that increased financial incentives can increase the quantity, but not the quality.

[1] http://crowdsourcing.typepad.com

[2] https://www.mturk.com/

In contrast, for the task of collecting large scale tagged data, we strive to reduce the cost while maintaining even improving the tagged data quality. We study to improve existing incentive model for crowdsourcing. Is it necessary for workers to be paid by a task assigner? What if the goal of the task assigner overlaps with some third party's interests for the data tagging task?

2.1.2 Social Computing

Social computing includes technologies enabling online communications between humans in a social role, such as blogs, Wikipedia and online communities. As shown in Figure 2.1b, it solves a computation problem implicitly through an online social application. For example, Wikipedia solves the problem of gathering existing knowledge and formulating it as prose through a dynamic social process of discussion about the facts and presentation of each topic among a network of authors and editors [34]. In social computing, workers generate data mainly for social purposes, and any performed computation is just a by-product of the application.

The collaborative potential of the social webs is often used to obtain tagged data. These sites allow users to assign keywords (tags) to the data that are then used for indexing and retrieval purposes. For example, *Flickr*[3] is a web-based photo sharing and social media web site, which allows users to upload photos and to provide textual tags as a way to describe objects in each photo. Others may add comments and tags to the image as well. Social bookmarking and tagging sites such as *Del.icio.us*, *StumbleUpon* and *Ma.gnolia* have helped search engines to index sites faster and give more quality results by analysing the inputs about a site from the users. More computational techniques used in social computing can be found from the survey by King *et al* [35].

However, this "free tagging by all" results in extremely large number of bad keywords as well as over-annotation (too many keywords per image). Different people express the same concept differently. Existing studies [36] have shown that half of the tags generated by general web users on Flickr are unrelated and useless. Many of them do not describe visual contents but the context of the image such as location and time.

[3] www.flickr.com

If worker's motivation in crowdsourcing is money. The participant's motivation in social computing is love and glory. Examples of love motivation include: intrinsic enjoyment of an activity such as games, socializing with other, and feeling of contributing to a cause larger than themselves. Workers in open source software communities are mainly motivated by glory, or called recognition. As Malone *et al* [37] pointed out, appealing to Love and Glory may reduce costs. Amazon does not pay for the book reviews; users write them to gain recognition or because they simply enjoy doing so. However, reliance on love and glory does not always work:

> "It is often difficult to control how fast or in what direction a crowd works. But if there are specific goals in mind, the crowd can sometimes be influenced to achieve them faster by providing money or glory to the members of the crowd who go in the desired direction. "

In this chapter, we study how to design a social application for collecting large scale tagged data at low cost as a task designer? The goal of SMC model is to provide a way that can harness the large number of participants in social webs using a monetary incentive provided by some third parties to control what direction a crowd works. To the best of our knowledge, there is not existing work on integrating the monetization in social computing for data collection purposes.

2.1.3 Human Computation

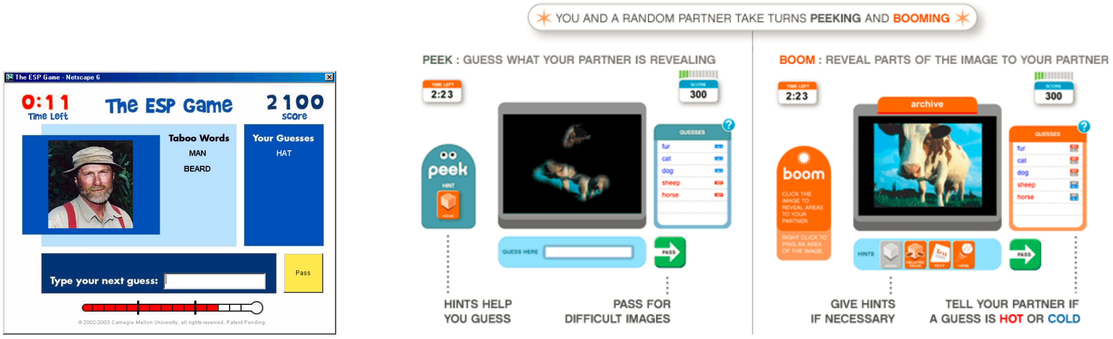
Human computation or Games with A Purpose (GWAP) [38] is first introduced by von Ahn's PhD dissertation. It is an innovative idea that makes use of human brain power to solve problems that computers cannot yet solve. As shown in Figure 2.1c, human computation solves a computation problem implicitly through an application serving users for a different purpose. For example, ESP game [21] solves image labeling problem through people play games for fun. ReCAPTCHA [39] solves character recognition problem via web security measures which prevent automated programs from abusing online services.

Many innovative approach to collect tagged images by unpaid workers were proposed in this domain. ESP game [21] aims at making the image annotation task enjoyable. As shown in Figure 2.2a, two players access the ESP game server and are randomly paired. A total of 15 images are shown to both players, and the goal of the game is for both players to type the same keyword for an image in order to win points. This scheme avoids the creation of bad and useless keywords. This game was later licensed by Google Image Labeler[4]. The Peekaboom game [40], shown in Figure 2.2b, is another similar game by Ahn *et al.* In addition to annotation, they also released to the public an image search engine based on the extracted annotated keywords. However, the keywords entered are usually at the image level. It was observed that as users are trying to win the game, many of them cheat, therefore creating low quality tags. Also, only general tags such as caption and location data were captured by the game which is only a small subset of the tags necessary to describe an image. Inspired by von Ahn's work, there is a large cluster of work relating to Games With A Purpose (GWAP) [21, 40–42]. Just for tagging image purpose, researcher proposed many different games, including Polarity [43], ShotoSlap [44] and KissKissBan [45].

Human computation is very similar to social computing. Both solve a computation problem implicitly through an application, except that social computing involves not only workers interacting with the application, but also participants interacting with each other in a social behaviour mediated by the application. From Figure 2.1b and Figure 2.1c, one can see that social computing model is an extension to human computation.

Because human computing solves problems by people contributing answers voluntarily, it also suffers from low quality data problem as mentioned above. For example, image labels generated by ESP game is at the image level instead of more detailed region level. People may also strive to win the game by cheating. In addition, human computation cannot collect the data at the same scale as social computing. According to Alexa [46], the top five global sites that have the highest Internet traffic are social networking or related sites.

[4] http://images.googlecom/imagelabeler/



(A) ESP game

(B) Peek and Boom

FIGURE 2.2: Some examples of collecting tagged images using human computation.

2.2 Social Monetization Computing (SMC) Model

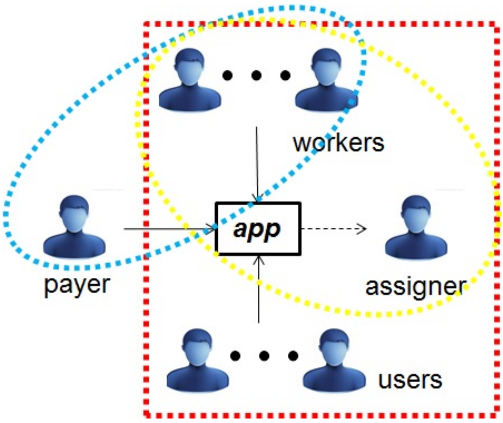


FIGURE 2.3: SMC model: a combination of crowdsourcing model (in blue area) and social computing model (in red area). Yellow area is human computation model. Better viewed in color.

We propose a Social Monetization Computing (SMC) model, which combines the crowdsourcing and social computing model, as shown in Figure 2.3. By introducing a payer to social computing, SMC model combines the motivation of crowdsourcing, which is monetary rewards, and the motivation of social computing, which is social communication. The model is based on the observation that many social behaviours can only be stimulated by generation of data in an online social community, and a third party who benefit from the social communication is willing to pay for such social behaviour, which implicitly pays for the generation of data as well. For example, advertisers and commercial companies are willing to pay for visitor traffic re-directed from other websites, which requires correctly linking (a kind of tagging) web information. Companies are willing to pay for popular reviews in Glassdoor in order to recruit prospective employees, which requires generation of review data.

In the SMC model, the task assigner needs to design an application that uses the generated data (in a form useful for the task assigner) to serve for a social communication purpose. Workers who contribute data are the "customers" of the application. Users are the group of the people who use the data in the social communication process through the application. Advertisers are "partners", because they would like to provide cashflow for the benefits provided by the system.

The SMC model is designed to ensure high quality and to reduce cost for the collecting data task. It benefits from worker's motivation, user-worker social interaction, and payer's interests in social interaction. Rather than generating free form data in social computing, workers are motivated to generate more data in high quality that is useful for a pre-defined social purpose. As shown in Figure 2.4, the larger number and the higher quality the data is, the higher chance the data is used in a social interaction, which leads to more payment to workers who generate the data. When designed properly, the collected data in SMC model is also useful for a task assigner who may alternatively use crowdsourcing to get the same data. For the task assigner, data collected in SMC model is at low cost, because there is a third party paying for the social behaviour, which implicitly pays the data as well. The data is also high quality, because workers strive to make more money from generated data, which depends on its usefulness in a social behaviour.

The SMC model can be applied to data collection problems that satisfy following rules:

- ***The collected data can stimulate a social behaviour.***
- ***There exists a third party who is willing to pay for such social behaviour.***

In order to apply SMC model to a data collection problem for a task assigner, one needs to answer the following questions:

- ***What is the targeted data?***

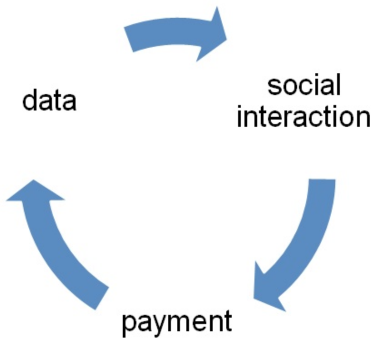


FIGURE 2.4: Dependency cycle of SMC model.

- ***What is the social behaviour that can generate such data?***
- ***What are the user roles in the social behaviour?***

Sometimes, the third party and the task assigner may be the same person. In such case, with the same cost, the task assigner achieves two goals: benefiting from the social interaction, and collecting useful data. Therefore, the cost for collecting useful data is reduced in the disguise of social interaction. For example, mobile-device vendors such as HTC and Motorola needs to tag bugs for a bug reports system [47] in order to better identify the root cause (Android platform, customized device-specific Android versions or customized software), and to provide swift and sufficient support/updates to their customers for better user experiences. By applying SMC model, these companies may save cost for tagging bugs. Instead of asking workers from crowdsourcing platforms to tag bugs, they can encourage their customers to tag when reporting bugs. Correctly tagged bugs (approved by vendor's developers) will be rewarded. In such way, the vendor not only collects wanted data, but also interacts with their customers in a social behaviour to gain customers' trust and feedback. The vendor may also save cost for software testing and supporting, since customers are willing to take an active role in the bug reports system.

2.3 A Case Study: Utilizing Image-Click-Ads Framework for Image Tagging

In this section, we present a case study of applying SMC model to the problem of collecting tagged image data. With the exponential growth of web image data, image tagging is becoming crucial in many applications such as e-commerce and content-based image retrieval. Collecting high quality tagged images at region level accuracy, e.g. mapping between content keywords and image regions, is crucial for many vision related techniques such as automatic object detection. Many tools (e.g. LabelMe [48], Markup SVG [19]) and related techniques are studied towards collecting large scale tagged image data such as ImageNet [14] at low cost. However, almost all of them are designed for the crowdsourcing scheme, and the tagging task is hard to be generalized for image data at web scale at an affordable cost.

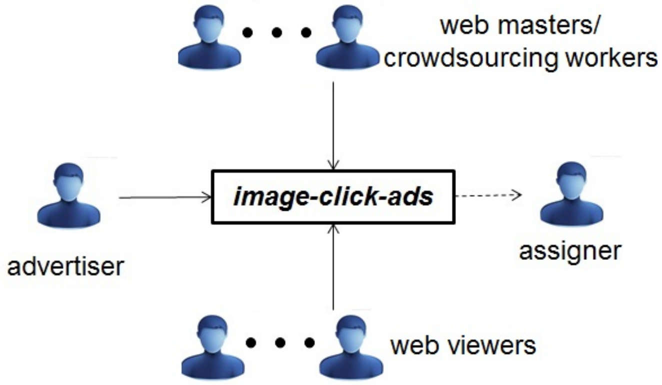


FIGURE 2.5: SMC model for collecting tagged images.

As shown in Figure 2.5, in order to solve the problem of collecting tagged image data by applying SMC model, three above mentioned problems are answered as follows:

- **The targeted data:** *tagged images should include content description and the corresponding region.*
- **The social behaviour:** *image-click-advertisement (image-click-ads) can be stimulated by tagged images.*
- **User roles:**

- **Workers:** web masters who want to monetize web images are workers; crowdsourcing workers who want to earn extra rewards through image-click-ads in addition to task rewards can also be workers;
- **User:** web viewers who want to get related information regarding image contents are users;
- **Payer:** advertisers who benefit from linking image contents to their products are payers.
- **Assigner:** researcher who need to collect large scale tagged images is the task assigner in this case.

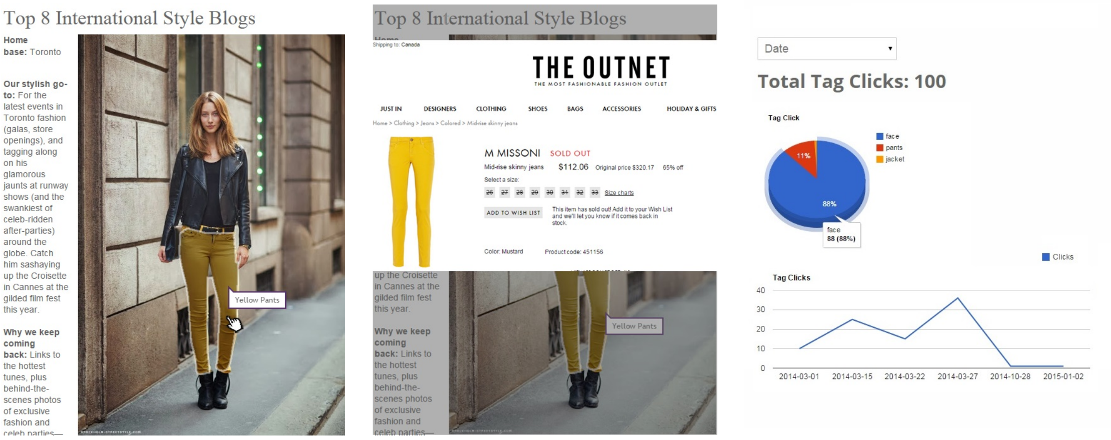
2.4 Evaluation

To better inform the design of our image tagging system, we conducted a series of formative studies investigating the SMC model in practice. Observations and feedback from these studies informed the design guidelines for data tagging systems using SMC model, as discussed in Design Guidelines for SMC Systems section. Following the design guidelines, an evolving prototype EyeDentifyIt 2.0 is presented in Chapter 3.

2.4.1 EyeDentifyIt 1.0

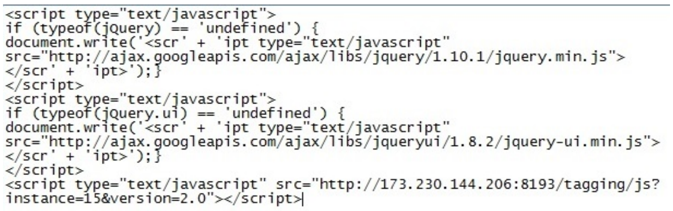
We conducted a preliminary study using an early prototype of our image tagging tool, named EyeDentifyIt 1.0, to examine the impact of SMC model on worker's motivation and data quality. We also collected feedback about how the tool could be improved. This prototype automatically links tagged image contents to commercial websites based on the content descriptions provided by workers, turning tagged contents into in-image advertisements, as shown in Figure 2.6a- 2.6b. By integrating Viglink[5] which provides APIs for revenue generating hyperlinks (from advertisers) and tracking web user interactions, the prototype provides a tracking report to each tagged image in order to track worker's earnings by image-click-ads. In addition, this prototype can be deployed in any website by installing our JavaScript code, as shown in Figure 2.7. The tagged images can be easily shared (with in-image advertisements) in different platforms, and can

[5] http://www.viglink.com/



(A) (B) (C)

FIGURE 2.6: An early prototype turns tagged objects in images into in-image advertisements. (a) Mouse over an image highlights the tagged object with the tag description; (b) Clicking the tagged object opens a store web site that contains a similar commercial item; (c) Clicking the tracking report (from our web dashboard) displays the interaction history for all tagged objects in an image.



```
<script type="text/javascript">
if (typeof(jQuery) == 'undefined') {
document.write('<scr' + 'ipt type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1.10.1/jquery.min.js">
</scr' + 'ipt>');}
</script>
<script type="text/javascript">
if (typeof(jQuery.ui) == 'undefined') {
document.write('<scr' + 'ipt type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.8.2/jquery-ui.min.js">
</scr' + 'ipt>');}
</script>
<script type="text/javascript" src="http://173.230.144.206:8193/tagging/js?
instance=15&version=2.0"></script>
```

FIGURE 2.7: The prototype generates JavaScript for our customers who want to monetize their web images. The JavaScript works in a similar way to Google Ads.

be easily retrieved in image search engines by keywords included in tag descriptions. For example, when workers tag an item using “yellow pants”, the tagged image can be retrieved when people search “yellow”, “pants” or “yellow pants”.

2.4.2 Comparison between SMC and Crowdsourcing

We created an image tagging task for $0.05/task on MTurk. The task contains two pars. In the first part, workers complete the task just like a common crowdsourcing task. The task requires a worker to select at least five images (from fashion image dataset Fashionista [4]) to tag fashion items, e.g. “tights”, “shorts”, “top”, using the public domain image tagging tool LabelMe [48], which requires to delineate an object region by clicking polygon vertices around the object's contours and to input the content description. Workers need to put their WorkerId in tag attribute field in order to track their work. In the second part, which is a follow-up questionnaire, workers were then informed the usage of generated data enabled by our prototype, which turns tagged images into live in-image advertisements. Workers can be paid extra bonus from image-click-ads revenue, e.g. \$0.01/click[6] for each tagged object (WorkerId in the tag attribute field is used for tracking earnings). The higher quality the tag is (accurate region outlines and tag descriptions), the higher chance that the tagged images will be used in different websites and will receive more clicks from web viewers. 35 workers including 18 females were recruited to finish the task.

In the first part, worker's motivation and data quality in a common crowdsourcing task (without SMC model) were examined. It revealed that workers are reluctant to tag more images than required without extra task rewards and creating highly accurate tag region is considered not necessary. On average, each worker tagged 5.11 images, slightly more than required. This suggests that most workers will not voluntarily invest more time than necessary to complete the tedious task of image tagging. Only 3/35 workers agree to tag more images without extra task rewards, mainly because "this was actually pretty fun" or "I just enjoy trying new task". Other workers gave the reason "I would consider doing more if there was a form of compensation" or similar ones. Only 2/35 workers agree to improve the tagging quality without extra task rewards, mostly because "I try to do hight quality tagging no matter what". Other workers gave the reason "work deserves rewards" or similar ones. These questionnaires reveal that worker's incentive during an image tagging task is affected by multiple factors including task rewards and how easy the task is, which is not surprising. In addition, their interests to improve tagging afterwards was mostly dependent on the task reward and was considered very low.

In the second part, after being informed image-click-ads function for tagged images enabled by our system, 33/35 workers expressed their willingness to tag more images given the same task reward, because "nice pay without extra work". 25/35 workers said

[6] \$0.01/click is a common practice in image-click-ads community according to Viglink

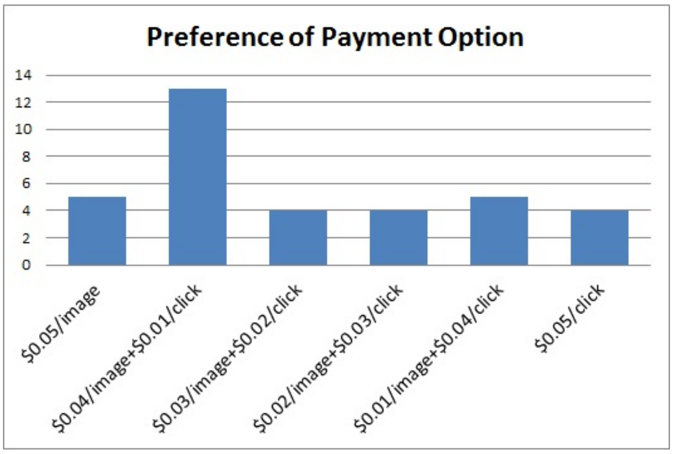


FIGURE 2.8: Distribution of crowdsourcing workers on preference of payment.

"yes" to the question "would you like to consider tagging more carefully than you just did without extra task rewards", mainly because "it may bring potential revenue", "the pay potential is rather greatly increased, I'm much more willing to tag more and better." The preference for different payment plans was collected from the questionnaire. As shown in Figure 2.8, some workers even prefer to be paid by image-click-ads revenue alone ($0.05/click). According to our payment survey, the cost of SMC model for image tagging task is reduced by 41.7% comparing to original cost ($0.05/image) in crowdsourcing model.

We also collected comments on possible improvements of the system. Most comments focusing on manual tagging tool in LabelMe site:

> "[Need] Better design of the LabelMe site, its hard on the eyes"

Further, participants felt that some automated techniques are needed to help tagging process:

> "Polygon tool needs to be finer. Something like the lasso tool from PhotoShop would be nice."

According to these comments, a semi-automated tagging tool is implemented in our final prototype to alleviate the manual labour.

2.4.3 Comparison between SMC and Social Computing

Web publishers who want to monetize their web images on their blogs/websites are typical users of image-click-ads platforms, e.g. Thinglink[7]. This group of users, unlike MTurk workers, are self-motivated to tag images without initial tagging payment. However, traditional image-click-ads platforms like Thinglink only allow workers to add intrusive interactive elements (such as animated dots) on top of images. Such tagged images are not useful for the task assigner-researcher in this case. One of our design goals is to provide a tool that allows web publishers to tag images useful for machine learning algorithms, while keeping web publishers as satisfied as using other image-click-ads platforms.

19 web publishers (age ranging from 20 to 49) were recruited to tag images using the semi-automated tagging tool. Feedback on satisfaction and how to improve the tool were also conducted. According to our interviews, although sixteen participants like the advertisements (tagged objects) overlaying images non-intrusively, all nineteen web publishers complained about the complexity of the semi-automated tagging tool. Typical answers were "It seems a lot of work." They requested to simplify the tagging tool that is good enough for in-image-ads capabilities. Nine also expressed the desire to automate certain process to improve the tagging efficiency-in one participant's own word:

> "Is there any way to automate some tagging part? "

Some web publishers also raised the concern of doing duplicate tagging work, because some images may have been tagged because of popularity:

> "If someone has tagged this image, is there any way to automatically retrieve the existing tags and modified according to my needs "

According to these comments, an automated tagging tool is implemented in our final prototype to reduce the workload for tagging each image. In addition, a duplication detection technique is implemented to avoid duplication work.

[7] https://www.thinglink.com/

2.5 Design Guidelines for SMC System

According to our findings form our user studies, SMC model provides better user motivation and data quality comparing to crowdsourcing model. However, regardless of user motivation and data quality in different model, users always desire less workload. If the SMC model can integrate better automated techniques to reduce the workload, it will provide better user experiences than crowdsourcing and social computing.

We summarise our findings as design guildlines for SMC system (for collecting large scale tagged data):

- Provide better user motivation by applying SMC model:
  - Define the targeted data.
  - Define the social behaviour that can generate such targeted data.
  - Define user roles in the social behaviour.
- Provide better user experience by reducing workload:
  - Reduce workload vertically for each tagging task by integrating automated tagging techniques.
  - Reduce workload horizontally between workers by integrating duplication detection technique.

2.6 Summary

With a series of formative studies, it is demonstrated that SMC model provides better user motivation to create high quality data for less cost than crowdsourcing model. We then apply SMC model to implement an early prototype of an image tagging system. When participants from crowdsourcing platforms use the early prototype, they requested to reduce the tagging workload of LabelMe. In addition, when participants from social webs use the early prototype, they requested to future reduce the tagging workload of semi-automated tagging tool. According to these feedback, SMC system design guidelines are summarised. In the next chapter, we will describe how the final prototype of the image tagging system is developed by following the design guidelines of the SMC system.

Chapter 3

EyeDentifyIt 2.0: Reducing Workload

According to our formative studies investing the impact of SMC model in practice, we developed the design guidelines for a SMC system. Following the design guidelines, an evolving prototype of the image tagging system is implemented. In this this chapter, we present the implementation details of this version of prototype, named EyeDentifyIt 2.0.

We describe our semi-automated and automated tagging tools and duplication detection technique for our final prototype in Section3.2. Our design decisions are related back to findings from our user studies. Section 3.5 explains the implementation of the tag-based image retrieval, which is used in the quality control for data collection. Experiments are designed to evaluate the SMC system, compared to state-of-the-art crowdsourcing system and social web system.

3.1 Background and Related Work

3.1.1 Automatic Image Tagging Methods

Amongst all image annotation methods, auto-annotation attracts the most of attentions. In the past decades, researchers have proposed different methods to automatically assign relevant keywords to images. The initial motivation for automatic image annotation is to improve the image search quality. Image retrieval is a difficult task because it is hard to find the correspondence between image keywords and image regions.

Guillaumin *et al* [49] identified mainly three groups of methods: generative models, discriminative models, and nearest neighbour (model-free or data driven) methods.

Generative models usually treat annotations as a translation from image instances to keywords. Different models of image and text co-occurrences are proposed [50, 51]. For example, topic models [52–55] annotate images as samples from a specific mixture of topics, with each topic a distribution over image features and annotation keywords. Mixture models define a joint distribution over image features and keywords. Given a new image, these models compute the conditional probability over keywords given the visual features by normalizing the joint likelihood. Some [56] use a fixed number of mixture components over visual features per keyword. Some use training images as components over visual features and keywords [57–59].

Discriminative models [60, 61] mainly learn a classifier for each keyword from human-tagged training images, then tag new images with keywords of the class which they belong to. In this category of methods, tagged image samples are collected and represented with low level features, and a machine learning model can then be trained using the matching between the feature and semantic tag. For texture feature alone, a dozens of features have been proposed, such as Gabor Filter [62, 63], Wavelets [64], Gabor Wavelet, [62] and Texton [65]. Materka *et al* [66] provided a complete review of texture features. Features are fed directly into conventional classifier which gives a yes or no results. Different learning models have been proposed, including support vector machines [67, 68], artificial neural networks [69], decision tree, [70, 71] and their different variants. Deep learning algorithms which are mentioned in Chapter 1 belong to this category. To our knowledge, different variants of deep learning algorithms have produced the best recognition rate so far. However, the best current approaches in this category can only deal with 1000 or so single object classes [10]. This is still a long way from the estimated $30,000$ or so categories that humans can recognize [72]. Another disadvantage of this type of approach is that it does not consider the fact that many images belong to multiple categories. Therefore, image retrieval using such image annotation methods can miss many relevant images which are tagged with not the right keyword.

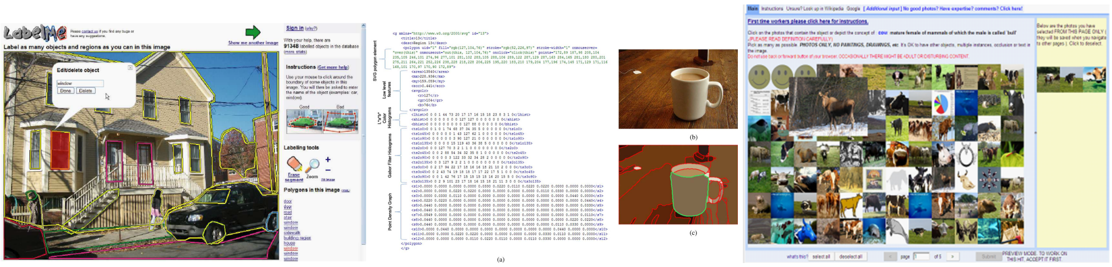
As the amount of training data increases, methods based on Nearest Neighbour (NN) becomes more effective. Examples include learning discriminate models using local similarities or perceptual distance between the query image and all training examples, named “SVM-KNN” [72], where K is the number of neighbours. Nearest neighbour of query images (defined in some feature space with a pre-specified distance measure) can also be used in image retrieval to transfer keywords [73, 74]. Data-driven approaches [75, 76] annotate images also belong to this category as it analyses textual information for similar images retrieved from web-scale search results. Such methods are effective for near-duplicate images but failed in tagging new images that do not have similarity matches in the data set.

Other than different learning models, researchers also made efforts on alternative data sources that can help automatic tagging. For example, Tsikrika *et al* [77] proposed to use click-through data acquired from the Web to train classifiers.

Current automated content description and annotation algorithms under development produce results that are very far from the level of detail a human annotator would do. For example, ImageEVAL [78] demonstrates algorithms generating global annotations have a higher success rate than algorithms attempting to detect specific objects. Particularly, algorithms distinguishing between city and landscape images, indoors and outdoors, are better than specific object detection, such as cars and sunglasses. Recognition of activities, events, and abstract or emotive qualities are even more difficult. In addition, most computer vision applications that have resulted from automated technology are strictly tied to the particular types of data set used. For example, a facial recognition algorithm is usually designed around the data set of real-world faces/objects used to train and test for it. Such trained model will not easily work for character faces in games, because the training data sets are not drawn or modelled characters. Most vision recognition tasks fundamentally reply on the ability to automatically recognize different object classes, in specific cases besides the resolution of general categories which in turn depends heavily on annotation and collection of training data set. For example, labeled images of faces and monsters from games are needed for training and recognizing game faces, which are very different from what normal facial recognizers are trained to expect. Many studies have shown that the accuracy of learned models depends on the quantity of the training images used and the ratio of the negative to positive images samples. The general rule of thumb has been that the larger the training set is, the more accurate the recognizer will get in identifying new instances of the same objects. Nonetheless, as stated before, the quality and quantity of the training data sets are the fundamental of a successful learned model [15], which makes the image tagging platform and the collecting strategy the key to solving the general image understanding problem.

Comparing to these methods, the proposed system tries to collect tagged images for training purposes. This requires that tagged images must be reliable, accurate, and at large scale. Currently only manual techniques can obtain reliable tags.

3.1.2 Dedicated Image Tagging Tools



(A) LabelMe

(B) Markup SVG

(C) ImageNet

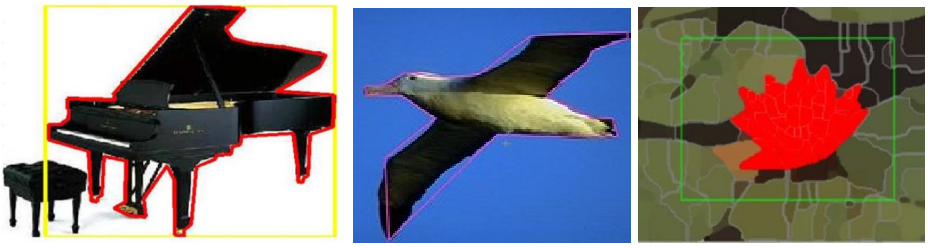
FIGURE 3.1: Some examples of tagging tools.

In the early days of computer vision, researchers started to collect data off-line by manually tagging images, e.g. Caltech 101 [79]. At that time, they usually tagged images using specialized programming tools such as Matlab, where tagged images were saved in .mat format. As the Web 2.0 technologies becomes more popular, it encourages new ways for enrolling people to perform the image tagging task. Several web-based tools dedicated to image tagging were developed to tag images using online communities. For example, LabelMe [48] is an on-line tool aiming at collecting keywords describing image regions for training and evaluating object recognition techniques. As shown in

Figure 3.1a, the user needs to click on the polygon vertices delineating an object's contour to generate a region. The incentive to annotate images is that the user can download the annotations. Markup SVG [19], as shown in Figure 3.1b, utilizes image processing technologies to help to tag object regions. However, it needs an image abstraction layer which is not easy to master by novice users. There are also tools specially designed for crowdsourcing platforms such as Amazon Mechanical Turk where workers can be hired to tag images, e.g. ImageNet tagging interface as shown in Figure 3.1c.

Such dedicated tagging tools can usually produce high quality tags. Figure 3.2 shows some examples of produced tagged images. However, there also exists many problems associated to those tools. First, general web users lack incentives to use these tools. Users are either research groups or workers hired by crowdsourcing companies to perform these tasks. It usually takes highly skilled labour to tag images of complex objects. Second, the development of such tools generally does not utilize image processing and/or recognition technologies, except for Markup SVG [19], which requires a special predefined abstraction layer that limits its usage by non-professional users.

3.1.3 Tagged Image Data Sets



(A) Caltech (B) LabelMe (C) Markup SVG

FIGURE 3.2: Sample images and tags from different systems. (a) Caltech 101 has ground truth annotations in Matlab format; (b) LabelMe is online with manual drawing polygon contours; (c) Markup SVG finds the object region through a specially defined image abstraction layer.

Tagged image data sets are widely used as ground truth in computer vision, such as object recognition, detection, and image annotation. Such data sets are not only useful for testing supervised learning algorithms, but also necessary to quantitatively measure their performances. Table 3.1 lists some major image data sets one can find in the literature. A more comprehensive survey about ground truth data sets can be found in Krig [80].

In the early days of computer vision, images were tagged manually by researchers. Some databases were limited in the number of classes (Caltech 101 [79], Caltech 256 [81]), some only provided image level tags and did not have detailed region representation (Corel 5K [55], PASCAL VOC [11]), some had specially defined tag formats that needed to be decoded (MSR Cambridge [82]). These data sets developed in the early days were not open to the public and could not be easily scaled-up in size. LabelMe [48] was the first attempt to make image tagging open to public contributions. This initiative resulted in over 30,000 tagged images with hundreds of categories and with a wide and comprehensive range of image selections. It also provided more detailed annotations, where individual object were marked by polygonal lines outlining its boundary. However, because general web users did not have incentives to label images, almost all users were limited to researchers. ImageNet [14] is so far the largest tagged image data set with over one million images. The data set is tagged by hiring workers from crowdsourcing platform like Amazon Mechanical Turk, which is expensive and hard to be generalized for large web usages. This data set only provides a bounding box annotation around each object instead of the more useful pixel level annotation contouring the object's boundary. One can summarize the problems that most datesets suffer from:

- **Lack of label accuracy**: when tagging images, very often keywords are usually associated with images instead of individual regions, e.g. Caltech 101 [79]. Such tagging is not helpful for training machine learning algorithms;
- **Constrained assumption**: many data sets have assumptions on the tagged objects, and do not provide different tagged objects in a complex scenes, e.g. Caltech 101 [79] and ILVRC 2010 [10]. Such data sets collected with a particular purpose in mind cannot be used by algorithms that exploit context analysis [83, 84];
- **Small number of classes**: many data sets only contain a small number of classes, such as faces, cars, pedestrians, and street scenes;

- **Unique format**: uniquely designed tag format makes the data hard to be shared and/or integrated with other systems;
- **Limited application**: almost all data sets are produced from a top-down approach, where the targeted application (training algorithms) determines the type of tagged objects. These data sets are rarely used beyond the domain of machine learning;
- **Not open and dynamic**: data sets produced in house makes the image tagging process hard to augment or to crowd-source.

Ideally, one should develop a system to collect a large, accurate, and continuously growing data set of tagged images where novice web users can contribute. This requires that the data set must be open and dynamic. In order to do so, one needs to design a format of tagged images that is generally acceptable by all web browsers. Such format, not only makes tagged images easy to be shared and/or integrated with general web technology, but also brings additional attributes for image-click-ads, such as hyperlink, interaction analysis, visualization, and animation.

3.1.4 Interactive Image Discovery

To the best of our knowledge, most image tagging methods neglect the original motivation of image understanding, which is when a viewer pro-actively engage with an image on a web page he/she should be able to ask a simple question like "what is this in the image?". There exist quite a few commercial systems that attempt to solve this problem for advertising. For example, image tagging platforms like Luminate[1], Pict[2], and Thinglink[3], are targeted to convert web users' purchasing impulse. In these systems tagging is limited to adding intrusive interactive elements on top of images. They also provide tag tacking capabilities which allow them to analyze tag usage for advertising revenue generation. This ways of tagging and tracking are not useful for image retrieval, content understanding, and machine learning. In contrast, our system generates tags that are non-intrusive interactive regions on top of images, which enables learning

[1] http://www.luminate.com/

[2] https://pict.com/

[3] https://www.thinglink.com/

TABLE 3.1: Comparison of existing tagged image data sets with EyeDentifyIt

Dataset	# of Classes	format	Accuracy Level	Methodologies	Open and Dynamic	Additional Attribute
Caltech 101 [79] Caltech 256 [81]	101/256	Matlab	outline polygon	in house	No	N/A
MSR Cambridge [82]	23	color coded image	pixel	in house	No	N/A
Corel 5K [55]	50	image	image level	in house	No	N/A
PASCAL VOC [11]	20	Matlab, XML	bounding box	in house	No	N/A
LabelMe [48]	free text	XML	outline polygon	crowd sourcing	Yes	N/A
ImageNet [14]	1000	Matlab	bounding box	crowd sourcing	No	synset
ESP [21]	free text	doc	image level	web game	Yes	N/A
Fashionista	53	JSON	superpixel	crowd sourcing	No	skeleton
EyeDentifyIt	free text	HTML Image Map	outline polygon	crowd sourcing	Yes	hyperlink interaction analytics visualization animation etc

from tagged object libraries where interaction analysis for objects inside images can be performed.

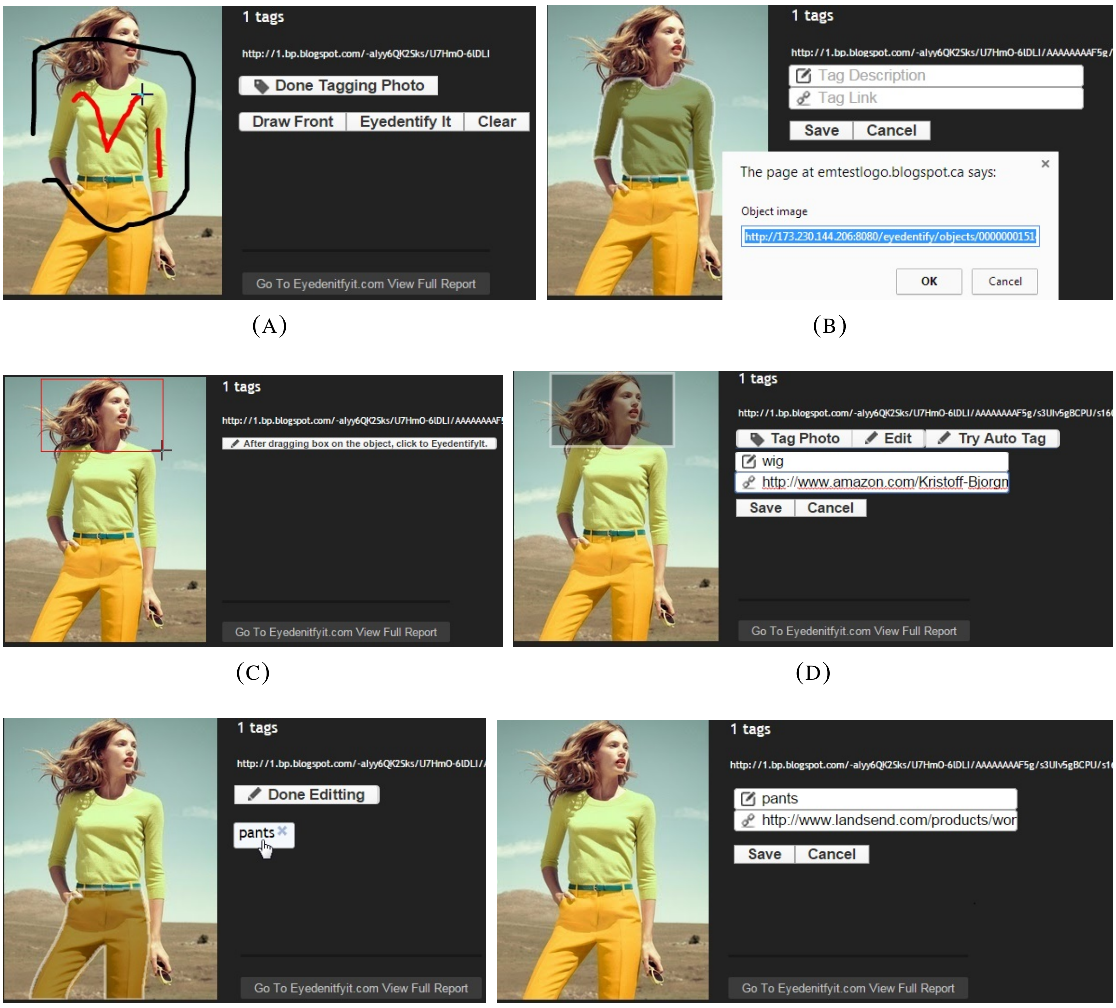
In summary, existing commercial in-image-ads tagging platforms are obtrusive and intrusive. They also do not use any automatic computer vision technologies to help the tagging process. In contrast, our platform can tag images in a non-intrusive way and can be partially automated using advanced image processing and recognition techniques.

3.1.5 Summary

Many data sets published by various researchers can be found at various diversity of objects, accuracy, and scale. However, none of them can be easily increased by and shared with general web users. In addition, neither of them can be easily integrated with general web technologies that allow sharing, editing, distributing, and permitting a contribution to the data set. Different annotation tools are proposed for collecting tagged images. Almost all these tools are designed for in-house tagging by researchers (e.g. Matlab labels Caltech 101 [79]), web-based tagging by dedicated researchers or crowdsourcing workers (e.g. LabelMe [48], Markup SVG [19]). None of them have considered the use of financial incentives and advanced computer vision techniques to assist the image tagging process in image-click-ads scheme. Crowdsourcing scheme currently provides the major human resource for large scale image tagging tasks. However, crowdsourcing has the problem of high cost and poor quality control. In our system, we proposed to change the existing incentive models of crowdsourcing by developing a new model to reduce image tagging cost and to improve its quality. In this scheme, workers can be rewarded financially to create high-quality tags by allowing better tagging revenues.

3.2 Vertical Workload Reduction

This section introduces the image tagging tools integrated in the prototype for vertical workload reduction. Unlike other fine-grained manual tagging platforms, e.g. Labelme [48] and crowdsourcing utilities [18], which require users to manually delineate boundaries or boxes around objects, EyeDentifyIt utilizes various techniques to alleviate the manual labour involved in the tagging process. To the best of our knowledge, amongst existing tagging systems, only Markup SVG [19] leverages image processing techniques to help in the tagging process.



http://1.bp.blogspot.com/-alyy6QK2Sks/U7HmO-6IDLI

The page at emtestlogo.blogspot.ca says:

Object image

http://173.230.144.206:8080/eyedentify/objects/0000000151

After dragging box on the object, click to Eyedentifylt.

wig

http://www.amazon.com/Kristoff-Bjorgn

pants

http://www.landsend.com/products/wor

(A)

(B)

(C)

(D)

(E)

(F)

FIGURE 3.3: A semi-automated tagging tool (a)-(b) is designed for crowdsourcing workers. An automatic tagging tool (c)-(d) is designed for web publishers. Existing tags are editable (e)-(f). (a) A worker marks lines (in red) on the object and lines (in black) on the background respectively. (b) The object area is automatically computed by clicking “EyeDentifyIt”. (c) A web publisher drags a bounding box around the targeted object. (d) A tag description “wig” and a tag link is auto filled. (e) Clicking “Edit” displays a list of existing tags. (f) Clicking “pants” allows modification of tag description and tag link.

3.2.1 Semi-Automated Tagging Tool

As observed from the usability study, crowdsourcing workers require better design for tagging tools like LabelMe. Therefore, we provided a new semi-automated tagging tool for crowdsourcing workers that is based on an image segmentation technique [85, 86].

As shown in Figure 3.3, a drawing tool is integrated into the JavaScript library so that a user is capable of marking foreground (in red) and background (in black) lines with respect to the region of interest (ROI). After marking lines, the user can trigger the “Eye-DentifyIt” button and then visualize the object segmented from its background using a graph-cut based image segmentation algorithm [85]. The algorithm defines the process of image segmentation as a graph labelling problem, where a graph is constructed using nodes mapped from pixels and edges connecting adjacent nodes with the edge weight computed by the pixel similarity. The segmentation is computed as an optimal labeling solution $X = \{x_i\}$, $x_i \in [0, 1]$ (0 and 1 represent foreground and background respectively) by minimizing a Gibbs energy [87] $E(X)$ defined on a graph $\mathbf{G} = \langle \mathbf{V}, \mathbf{E} \rangle$:

$$E(X) = \sum_{i \in \mathbf{V}} E_1(x_i) + \lambda \sum_{(i,j) \in \mathbf{N}} E_2(x_i, x_y), \tag{3.1}$$

where $\mathbf{V}$ is the set of nodes, $\mathbf{N}$ is the set of edges connecting neighbourhood nodes in the edge set $\mathbf{E}$, E_1 and E_2 are the unary term and pairwise term respectively. The algorithm formulates and solves the energy minimization problem using a max-flow min-cut algorithm. Boykov *et al* provides a complete review of graph-cut based energy minimization algorithms [88]. The user can continuously modify the marking lines until the segmentation result is satisfactory. The polygon lines around the object boundaries are automatically computed from the segmented object region, saved in Scalable Vector Graphics(SVG) format first. A compressed map coordinates is then computed from SVG and saved in the format of HTML image map.

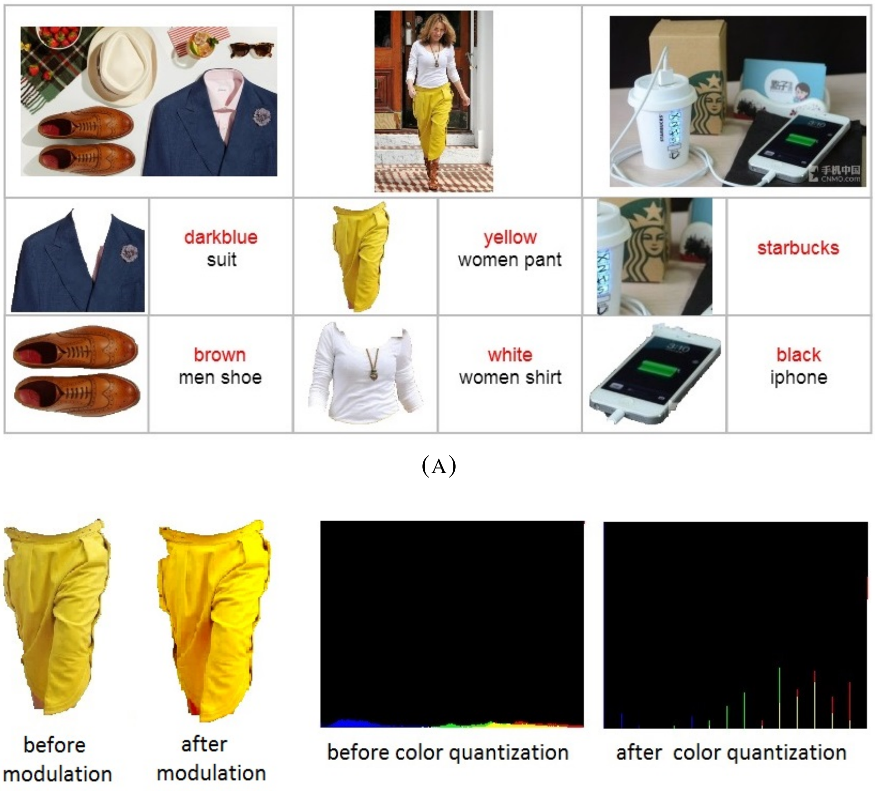
After tagging an object’s region in an image, the user is required to input a tag description and a tag link. The link provides the monetized advertisement link that is open in a new tab when the object is clicked. Meanwhile, the tagged object is saved in the object library. The tag description and the tag link are editable, as shown in Figure 3.3 (e)-(f).

3.2.2 Automated Tagging Tool

As indicated from the usability study, web publishers often encounter obstacles with the semi-automated tagging tool. Therefore, additional supports are needed to help web publishers to overcome these obstacles. These supports are included as part of the automated tagging tools and include:

A) Automatic Object Recognition Unlike MTurk workers who would like to invest time on learning the semi-automated tagging tool (because of the initial task reward incentive), web publishers often felt that it has barriers to use. Considering image tagging in the form of bounding boxes is necessary for many computer vision tasks, such as object detection, the system provides a bounding box tagging tool for the needs of web publishers. As shown in Figure 3.3 (c)-(d), a user only needs to drag a bounding box around the object area to tag an item. When releasing the mouse, the bounding box area is submitted to a machine-trained process to perform auto recognition. Currently, a pre-trained convolutional neural network model proposed by Krizhevsky *et al* [89] is used for auto recognition. The model is trained from the data set ILSVRC1K [90] where labels are synset organized by the WordNet hierarchy [91], and can recognise up to 1,000 classes. If the object is recognized at a probability higher than a pre-defined threshold, the recognized class label is auto filled in the tag description field. With the tag description, the tag link can also be auto filled through Viglink API. If the auto-recognition fails, the user can always chose to edit the auto-filled tag description and the link.

The benefits of integrating automated techniques here are mainly three folds. First, it automatically adds the tag description (via trained models) and the tag link (via Viglink API), which reduces worker's workload. Second, researchers can deploy automated techniques (e.g. trained models) in the system (modular design), and keep collecting the usage statistics, e.g. which automatically tagged regions are kept or deleted by users. Such data can be useful for researchers for reinforcement learning algorithm, which improves the automated technique. Third, tags generated by automated techniques can be verified by workers, and become useful data for other researchers who create new training data sets.



(A)

(B)

FIGURE 3.4: Examples of auto label computation. Better viewed in color. (a) Three examples of images (in the 1st row), tagged objects (sub images in the 2nd and 3rd row) and their labels. Labels in red are added by the system without manual interruption through auto label computation. Labels in black are added by users. (b) The color histograms of the pants before and after color quantization.

B) Other Automatic Label Computation In the usability study, many web publishers wish that the system could automatically add certain tags. As part of the assisted tagging tools, EyeDentifyIt automatically computes labels based on low-level image features such as: color and SIFT features. SIFT stands for scale-invariant feature transform, an image feature descriptor frequently used for image-based matching and recognition. The algorithm was published by Lowe [92] in 1999. It computes interest points (local features) by calculating statistics of local gradient directions from image intensity.

Examples of auto label computation are shown in Figure 3.4a. The color model employs image processing library ImageMagic[4] to analyse the dominant color of the tagged area and to generate a color tag in addition to user generated tags. The dominant color is computed using the dominant color from the RGB color histogram of the image after converting RGB into HSV color space (modulating its brightness and saturation) and color quantization for the region. Figure 3.4(b) shows an example of conversion and histogram output before and after color quantization. In the histogram, the x-axis is the color value (0-255) and the y-axis is the pixel count. The histogram for each channel is displayed in the color it represents. Thus, red and blue overlap to make magenta. In other words each color channel has its own separate histogram. In our system, SIFT features are computed at the image level to perform auto logo recognition. For each image to be tagged, its SIFT feature is computed and compared against a logo database. Random Sample Consensus (RANSAC) algorithm proposed by Fischler and Bolles in 1981 [93] is used to find the transforming matrix between the set of matching SIFT features. If a reliable transforming matrix is found between the query image and query logo, the logo tag is automatically added to the user generated tags. This is especially interesting for brand marketing purposes, e.g. Starbucks is interested to know how many users share images holding Starbucks coffee logo.

Overall, auto computed labels are implicitly added as part of the tag description and saved in the database without user's intervention. They are not visible to users in order to avoid confusion. However, they are used in search engine to help more accurate image retrieval, which is detailed in Section 3.5.

3.3 Horizontal Workload Reduction

During the usability study, some web publishers raised the concern of doing duplicate tagging. Not only for web publishers, this is a common yet unresolved problem for the image tagging task. Currently almost all image data sets are built from scratch. Examining weather an image (in the collected training data) has been tagged before by

[4] http://www.imagemagick.org/script/index.php

someone else and weather the tagged data can be reused probably cost more than simply tagging it again. However, such repeated work is costly and very inefficient.

```
input  : Image $Q$ of size $w \times h$
output: N-bit Hash code
1. Resize image to $32 \times 32$ and change to gray scale;
2. Compute 2-D DCT coefficient get $32 \times 32$ matrix A;
3. Keep the top-left $8 \times 8$ coefficient get matrix B, which represents the lowest frequencies (perceptually most significant) in the image;
4. Compute the mean DCT coefficient value from matrix B excluding the $B_{00}$ to eliminate influences of the commonly global variance of luminance in the photo.;
5. Compute the bits sequence: set each matrix bit depending on each coefficient is above or below the average DCT value. Order the bits into a 64-bit integer;
```

Algorithm 1: Proposed perceptual hashing

Images in EyeDentifyIt are indexed by uniform resource locator (URL). An image whose URL has not been stored in the system are considered as new. However, very often identical images on the web may have different URLs. An image with a new URL does not mean it has not been tagged before. For example, user A and B have the same set of images, but they have two different websites. When they post the images onto their own websites, each image has a unique URL. However, if user A has tagged any image on the website, then the identical image on user B's website should automatically retrieve the existing tags generated by user A. Two images are considered identical when they have the same visual content regardless of changes to scale, size, color, storage format etc. After retrieving the existing tags, user B can always chose to keep, modify, or delete the existing tags.

Typically, the similarity of two images is measured in Euclidean space. However, measuring pairwise Euclidean distance using high dimension features is generally not scalable and becomes a bottleneck when the search space is significantly increased. Here, we propose a perceptual hashing [94] algorithm to avoid repeated tagging problem. The algorithm is widely used to protect copyright infringement and content-based image retrieval/search [75]. Particularly, discrete cosine transform (DCT) based method is implemented in EyeDentifyIt. DCT based method works well for variations in scale, aspect ratio, colors, and storage format etc, but fails in a range of geometric transformations [94] such as rotation, reflection, and translation. Such property fits the system's need for automatic tag retrieval. Because most tags in the system are spatially extended tags (generated by manual tagging tool), therefore an image where geometric transformations was applied should not get the same tag region as before the transformation.

The proposed perceptual hashing algorithm for resolving repeated tagging is described in Algorithm 1. First, it transforms the color image into a smaller size gray scale image, represented by a matrix A. Two dimensional DCT transformation [95] is then applied to the image matrix A, and only the top left 8×8 coefficients which are the lowest frequencies representing significant structures of the image are kept for further computation, represented by matrix B. Since the B_{00} called DC coefficient is the average image intensity, we exclude it from the mean DCT computation to eliminate the influences of global variance of luminance in the image. Finally, we set the 64 hash bits to 0 or 1 depending on weather each of 64 DCT values is above or below the average value.

Each tagged image has an associated hash code saved in the database. For each new image (with new URL), the hash code is computed and compared against all hash codes saved in the database by counting the number of bit positions that are different using Hamming distance. Two images are considered identical if the Hamming distance between their hash values is smaller than a predefined threshold, which is selected experimentally to be 12 in our system implementation.

To the best of our knowledge, this is the first time in image tagging that the problem of repeated image tagging is identified and resolved. As the collected image data in the system grows, this hashing capability will become more important.

3.4 System Overview

3.4.1 Deployment

The current version of the system has been deployed as a publicly-available web application and hosted at: http://www.EyeDentifyIt.com. The web front end is developed using JavaScript, HTML, and PHP. The web application connects to a centralized web server, which is responsible for the following tasks: receiving requests from client applications, distributing requests to different processing modules, storing all the information, collecting tracking analytics etc.

The system dashboard interface is shown in Figure 3.6. Images from registered websites are automatically downloaded into the user's media library using an image crawler. All tagged images are listed under "tagged" tab. Similar to the design of Google Images, the selected image expands a tagging panel, which includes tagging widgets and a link to the tracking report summarizing the statistic such as image hover/clicks/sharing. A user can tag and track all images hosted on the registered web site from either dashboard or the registered website directly by signing-in the account (with sign-in hot key "SHIFT+L"). The same tagging panel can be displayed for the selected image when users sign-in from the registered website directly.

Unlike traditional image-click-ads interaction interfaces like Thinglink, which add intrusive elements on top of images, the ads-enabled tags in EyeDentifyIt are designed to add non-intrusive elements on top of images. As shown in Figure 2.6, the tagged area will only be visualized and interactive when the mouse is over a tagged region. Such design can keep the web pages that installed the JavaScript as clean as possible, so that web publishers who want to monetize their websites can feel free to use EyeDentifyIt without worrying that the tagged images change the appearance of the website.

3.4.2 Work Flow

Compared to image tagging systems in research community, our system is designed in a way to incorporate image tagging process in the image-click-ads scheme. Because of the nature of this design, the system needs to have three basic functionalities: tagging tools tag images in a form useful for the task assigner (data needs to be compatible with machine learning algorithms), tracking tools collects revenues from ads-enabled tags, learning tools learn better automated models from collected data.

Overall, the system is composed of three modules: tagging tools, tracking tools, and learning tools. These module classes are enabled on the registered websites through the JavaScript installation. Figure 3.5 illustrates the work flow of the system. Similar to

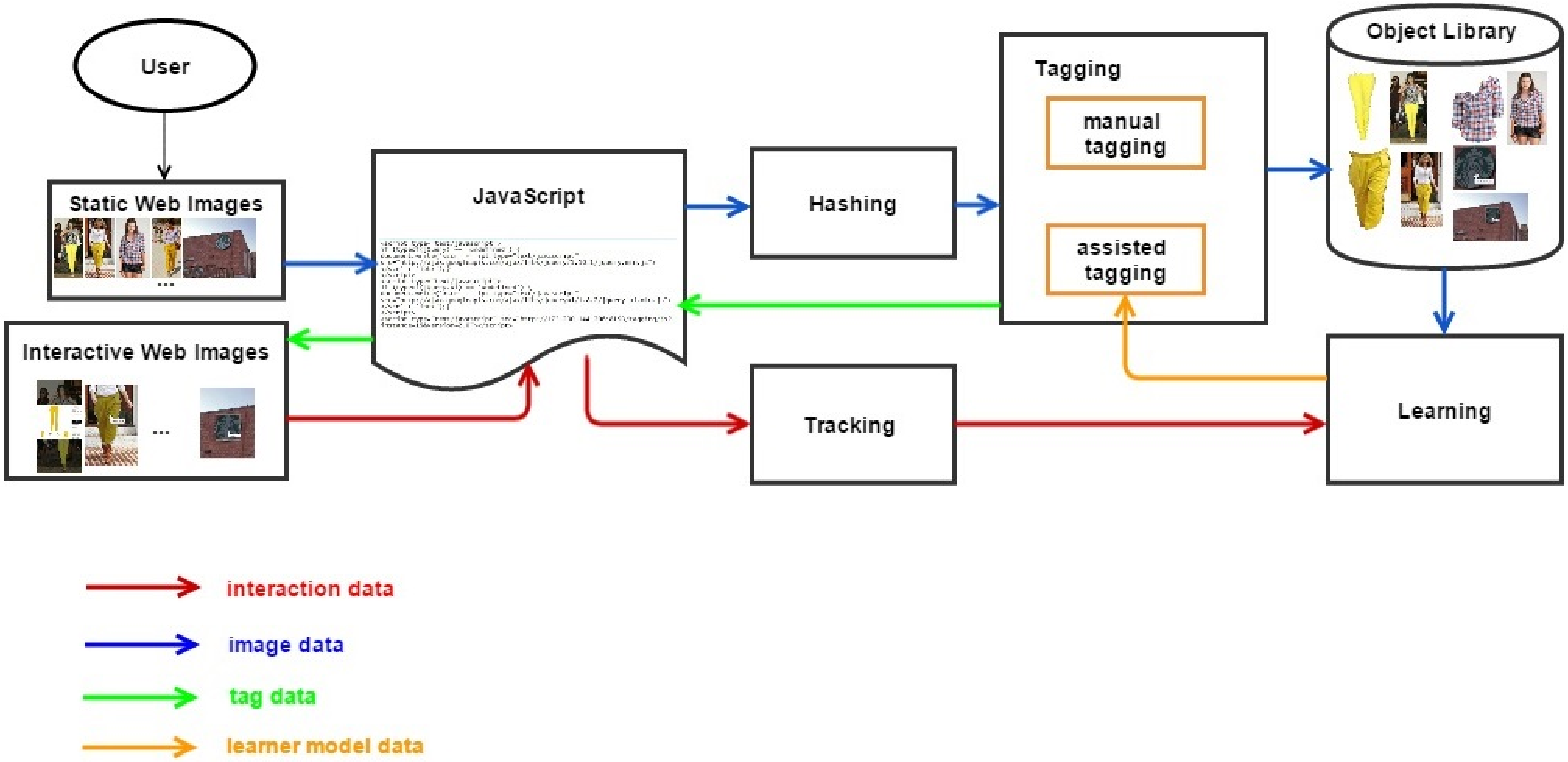
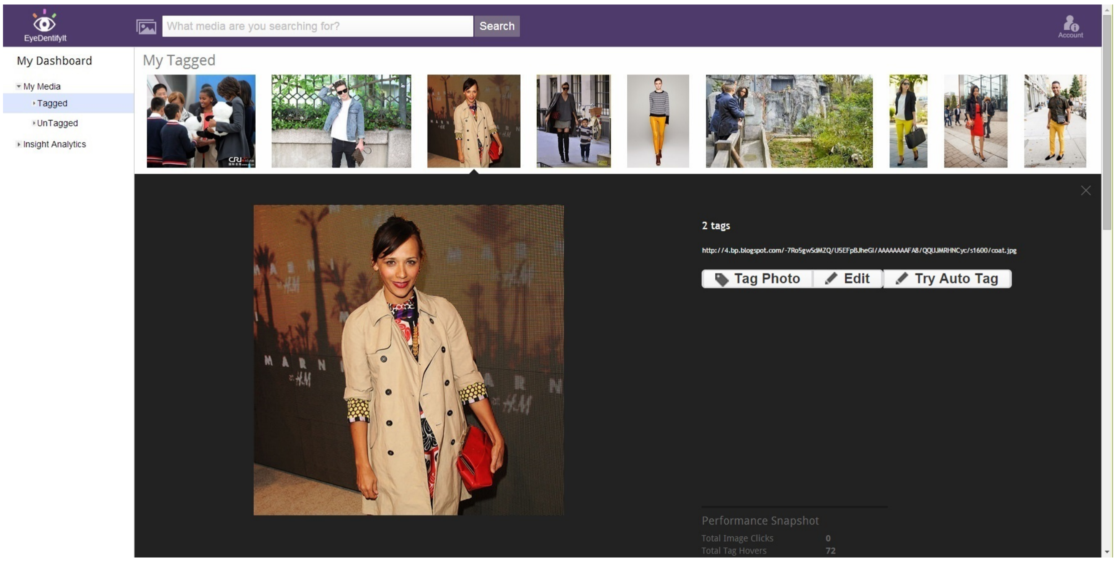


FIGURE 3.5: The flowchart of the system framework. The system mainly contains three modules: tagging, tracking and learning. Blue lines indicate the flow of the image data. Red lines indicate the flow of the tracking data. Green lines indicate tag data. Orange lines indicate learner model data.



2 tags

http://4.bp.blogspot.com/-7Ro5gwSdMZQ/U5EFpBJheGI/AAAAAAAAFA8/QQUJMRHNCyc/s1600/coat.jpg

Performance Snapshot

Total Image Clicks 0

Total Tag Hovers 72

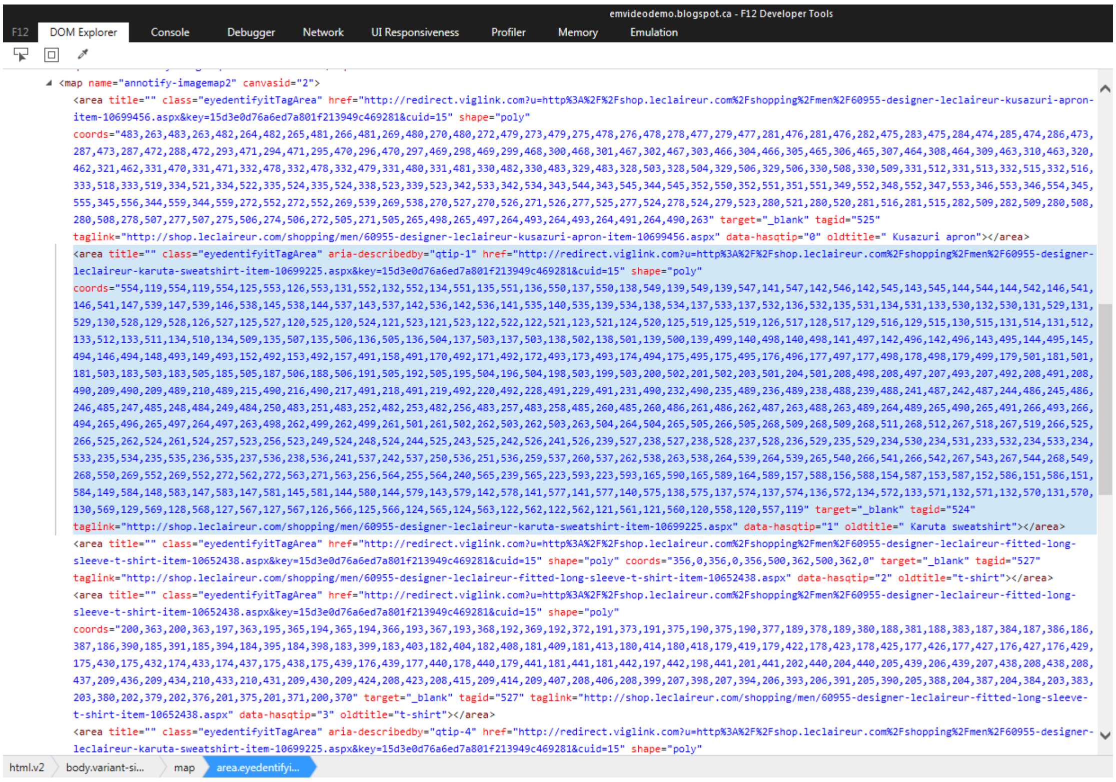
FIGURE 3.6: System dashboard: the selected image from user's tagged image library expands a tagging panel.

common image-click-ads platforms (e.g. Thinglink), for every sign-up user, the system generates a dedicated JavaScript code for the website. Once the JavaScript is installed (copy/paste) in the user's website, all images on the website have tagging and tracking abilities. Tagged images in the web browser possess interaction ability, and each interaction (left mouse click) opens another website (magnetizable hyperlink) in a browser's new tab. The tagging tools provide utilities for taggers to tag objects in images. Tagged objects are automatically saved in the object library. All the web interaction data are tracked through the tracking module via the JavaScript for revenue collection purposes. The learning module allows the system to analyse tagged images using machine learning techniques, which can then be used to develop better automated tagging tools.

In addition, in order to avoid duplicate work, each new images go through hashing first before worker starts to tag. If an identical image is found from the database, existing tags are automated retrieved to reduce workload.

3.4.3 Tagged Data Format

Ads-enabled image tags are designed as in-image advertisements which are visible, interactive, and shareable on the web. The HTML image map (containing polygon vertices delineating object's contour) is used as a data structure to store the tagged object



```
◢ <map name="annotify-imagemap2" canvasid="2">
    <area title="" class="eyedentifyitTagArea" href="http://redirect.viglink.com?u=http%3A%2F%2Fshop.leclaireur.com%2Fshopping%2Fmen%2F60955-designer-leclaireur-kusazuri-apron-
    item-10699456.aspx&key=15d3e0d76a6ed7a801f213949c469281&cuid=15" shape="poly"
    coords="483,263,483,263,482,264,482,265,481,266,481,269,480,270,480,272,479,273,479,275,478,276,478,278,477,279,477,281,476,281,476,282,475,283,475,284,474,285,474,286,473,
    287,473,287,472,288,472,293,471,294,471,295,470,296,470,297,469,298,469,299,468,300,468,301,467,302,467,303,466,304,466,305,465,306,465,307,464,308,464,309,463,310,463,320,
    462,321,462,331,470,331,471,332,478,332,478,332,479,331,480,331,481,330,482,330,483,329,483,328,503,328,504,329,506,329,506,330,508,330,509,331,512,331,513,332,515,332,516,
    333,518,333,519,334,521,334,522,335,524,335,524,338,523,339,523,342,533,342,534,343,544,343,545,344,545,352,550,352,551,351,551,349,552,348,552,347,553,346,553,346,554,345,
    555,345,556,344,559,344,559,272,552,272,552,269,539,269,538,270,527,270,526,271,526,277,525,277,524,278,524,279,523,280,521,280,520,281,516,281,515,282,509,282,509,280,508,
    280,508,278,507,277,507,275,506,274,506,272,505,271,505,265,498,265,497,264,493,264,493,264,491,264,490,263" target="_blank" tagid="525"
    taglink="http://shop.leclaireur.com/shopping/men/60955-designer-leclaireur-kusazuri-apron-item-10699456.aspx" data-hasqtip="0" oldtitle=" Kusazuri apron"></area>
    <area title="" class="eyedentifyitTagArea" aria-describedby="qtip-1" href="http://redirect.viglink.com?u=http%3A%2F%2Fshop.leclaireur.com%2Fshopping%2Fmen%2F60955-designer-
    leclaireur-karuta-sweatshirt-item-10699225.aspx&key=15d3e0d76a6ed7a801f213949c469281&cuid=15" shape="poly"
    coords="554,119,554,119,554,125,553,126,553,131,552,132,552,134,551,135,551,136,550,137,550,138,549,139,549,139,547,141,547,142,546,142,545,143,545,144,544,144,542,146,541,
    146,541,147,539,147,539,146,538,145,538,144,537,143,537,142,536,142,536,141,535,140,535,139,534,138,534,137,533,137,532,136,532,135,531,134,531,133,530,132,530,131,529,131,
    529,130,528,129,528,126,527,125,527,120,525,120,524,121,523,121,523,122,522,122,521,123,521,124,520,125,519,125,519,126,517,128,517,129,516,129,515,130,515,131,514,131,512,
    133,512,133,511,134,510,134,509,135,507,135,506,136,505,136,504,137,503,137,503,138,502,138,501,139,500,139,499,140,498,140,498,141,497,142,496,142,496,143,495,144,495,145,
    494,146,494,148,493,149,493,152,492,153,492,157,491,158,491,170,492,171,492,172,493,173,493,174,494,175,495,175,495,176,496,177,497,177,498,178,498,179,499,179,501,181,501,
    181,503,183,503,183,505,185,505,187,506,188,506,191,505,192,505,195,504,196,504,198,503,199,503,200,502,201,502,203,501,204,501,208,498,208,497,207,493,207,492,208,491,208,
    490,209,490,209,489,210,489,215,490,216,490,217,491,218,491,219,492,220,492,228,491,229,491,231,490,232,490,235,489,236,489,238,488,239,488,241,487,242,487,244,486,245,486,
    246,485,247,485,248,484,249,484,250,483,251,483,252,482,253,482,256,483,257,483,258,485,260,485,260,486,261,486,262,487,263,488,263,489,264,489,265,490,265,491,266,493,266,
    494,265,496,265,497,264,497,263,498,262,499,262,499,261,501,261,502,262,503,262,503,263,504,264,504,265,505,266,505,268,509,268,509,268,511,268,512,267,518,267,519,266,525,
    266,525,262,524,261,524,257,523,256,523,249,524,248,524,244,525,243,525,242,526,241,526,239,527,238,527,238,528,237,528,236,529,235,529,234,530,234,531,233,532,234,533,234,
    533,235,534,235,535,236,535,237,536,238,536,241,537,242,537,250,536,251,536,259,537,260,537,262,538,263,538,264,539,264,539,265,540,266,541,266,542,267,543,267,544,268,549,
    268,550,269,552,269,552,272,562,272,563,271,563,256,564,255,564,240,565,239,565,223,593,223,593,165,590,165,589,164,589,157,588,156,588,154,587,153,587,152,586,151,586,151,
    584,149,584,148,583,147,583,147,581,145,581,144,580,144,579,143,579,142,578,141,577,141,577,140,575,138,575,137,574,137,574,136,572,134,572,133,571,132,571,132,570,131,570,
    130,569,129,569,128,568,127,567,127,567,126,566,125,566,124,565,124,563,122,562,122,562,121,561,121,560,120,558,120,557,119" target="_blank" tagid="524"
    taglink="http://shop.leclaireur.com/shopping/men/60955-designer-leclaireur-karuta-sweatshirt-item-10699225.aspx" data-hasqtip="1" oldtitle=" Karuta sweatshirt"></area>
    <area title="" class="eyedentifyitTagArea" href="http://redirect.viglink.com?u=http%3A%2F%2Fshop.leclaireur.com%2Fshopping%2Fmen%2F60955-designer-leclaireur-fitted-long-
    sleeve-t-shirt-item-10652438.aspx&key=15d3e0d76a6ed7a801f213949c469281&cuid=15" shape="poly" coords="356,0,356,0,356,500,362,500,362,0" target="_blank" tagid="527"
    taglink="http://shop.leclaireur.com/shopping/men/60955-designer-leclaireur-fitted-long-sleeve-t-shirt-item-10652438.aspx" data-hasqtip="2" oldtitle="t-shirt"></area>
    <area title="" class="eyedentifyitTagArea" href="http://redirect.viglink.com?u=http%3A%2F%2Fshop.leclaireur.com%2Fshopping%2Fmen%2F60955-designer-leclaireur-fitted-long-
    sleeve-t-shirt-item-10652438.aspx&key=15d3e0d76a6ed7a801f213949c469281&cuid=15" shape="poly"
    coords="200,363,200,363,197,363,195,365,194,365,194,366,193,367,193,368,192,369,192,372,191,373,191,375,190,375,190,377,189,378,189,380,188,381,188,383,187,384,187,386,186,
    387,186,390,185,391,185,394,184,395,184,398,183,399,183,403,182,404,182,408,181,409,181,413,180,414,180,418,179,419,179,422,178,423,178,425,177,426,177,427,176,427,176,429,
    175,430,175,432,174,433,174,437,175,438,175,439,176,439,177,440,178,440,179,441,181,441,181,442,197,442,198,441,201,441,202,440,204,440,205,439,206,439,207,438,208,438,208,
    437,209,436,209,434,210,433,210,431,209,430,209,424,208,423,208,415,209,414,209,407,208,406,208,399,207,398,207,394,206,393,206,391,205,390,205,388,204,387,204,384,203,383,
    203,380,202,379,202,376,201,375,201,371,200,370" target="_blank" tagid="527" taglink="http://shop.leclaireur.com/shopping/men/60955-designer-leclaireur-fitted-long-sleeve-
    t-shirt-item-10652438.aspx" data-hasqtip="3" oldtitle="t-shirt"></area>
    <area title="" class="eyedentifyitTagArea" aria-describedby="qtip-4" href="http://redirect.viglink.com?u=http%3A%2F%2Fshop.leclaireur.com%2Fshopping%2Fmen%2F60955-designer-
    leclaireur-karuta-sweatshirt-item-10699225.aspx&key=15d3e0d76a6ed7a801f213949c469281&cuid=15" shape="poly"
```

FIGURE 3.7: Snapshot of tag data structure.

regions in an image. A JavaScript library was developed to allow the visualization of the tagged regions everywhere that the JavaScript is installed, including modern browsers, and mobile devices like iPads, iPhones, and Android phones. The tagged regions can be visualized, interacted with, and become content-aware by on-click, on-mouseover, on-mouseout etc. events. A snapshot of the tag data retrieved via the JavaScript is shown in Figure 3.7. The data structure contains tagID, shape (e.g. polygon), coordinates (describing the shape), href (monetized hyperlink via Viglink), taglink (advertisement link), and title (tag description).

3.5 Quality Control Using Tag-based Image Retrieval

A basic quality control strategy in crowdsourcing is to use majority voting, which is collecting multiple answers from different taggers and taking the consensus. This approach has been successfully used for many image tagging tasks such as verifying the presence of objects in images [14, 18]. However, drawing a tag region is significantly more difficult and time consuming than giving answers to a validation question, e.g.

weather the tagged region is accurate for the tag description. Thus instead of using multiple taggers to reach consensus, a new quality control strategy via tag-based image retrieval results is proposed. This new verification scheme is based on the following:

- **Search tag:** An image search action is initiated by searching the tag name that needs to be verified. For example, if the researcher wants to verify the tagged object "shirt" in all images, the system searches images using the keyword "shirt". In such a way, only images with correct spelling of the object name are retrieved, and images with incorrect tag name such as "shert" are filtered out;
- **Verify region:** For every image retrieved via the search keywords, a user needs to mouse over the region described by the keyword. With highlighted tag region, it is easy to examine weather the tagged region for the search keyword is accurate or not. In this scheme, the user needs to check each image in the search results and examine weather the tagged region for example "shirt" is accurate.

TABLE 3.2: Symbols and semantics defined for tag-based image retrieval

Symbol	Semantic
$\mathbf{D}$	The tagged image collection
d	A tagged image, $d \in \mathbf{D}$
$\mathbf{T}_d$	The tag collection associated with image d
$t_d \in \mathbf{T}_d$	A tag t associated with image d
$\mathbf{L}$	A label set serves as lexicon
$\mathbf{L}_t \subseteq \mathbf{L}$	The label set associated with tag t
$l_t \in \mathbf{L}_t$	A label l associated with tag t
$\mathbf{T}_l$	A tag collection associated with label l
$f(\mathbf{L}_{\lvert m\rvert})$	The set of tags associated with label set $\mathbf{L}_{\lvert m\rvert} = \{l_1, l_2 ... l_m\}$, $f(\mathbf{L}_{\lvert m\rvert}) = \bigcap_{i=1}^{m} T_{l_i}$
$g(\mathbf{T}_{\lvert n\rvert})$	The set of images associated with tag set $\mathbf{T}_{\lvert n\rvert} = \{t_1, t_2 ... l_n\}$, $g(\mathbf{T}_{\lvert n\rvert}) = \bigcap_{i=1}^{n} D_{t_i}$
$t_q \in \mathbf{Q}$	A query tag in query $\mathbf{Q}$

In this framework, both sub-tasks serve to control the quality of the generated tags. Meanwhile, since the 2nd sub-task only requires a binary answer, it is more time and cost efficient. The quality of the 2nd task can be easily controlled by well-proven techniques such as majority voting strategy.

The tag-based image retrieval is implemented by using tags to index images in the database, and to represent the combination of tags in a lexicon. The notations used in this section are presented in Table 3.2. Figure 3.8 depicts the process of tag-based image retrieval system. To allow users to do cross-tag (or multi-tag) image retrieval, a query can be defined in the format of $t_1\&t_2\&...\&t_q$, with “&” sign separating each tag description. The corresponding retrieved image sets $\mathbf{D}_{t_{|q|}}$ are then computed as :

$$\mathbf{D}_{t_{|q|}} = \bigcap_{i=1}^{q} g(f(\mathbf{L}_{t_i})). \tag{3.2}$$

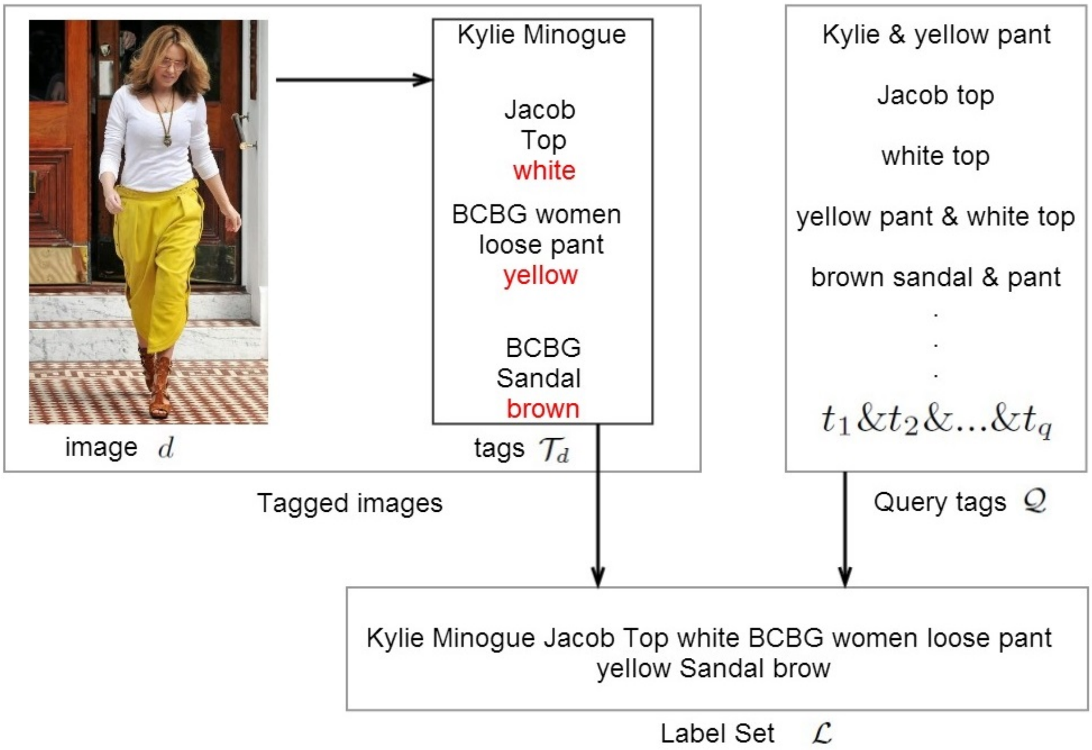
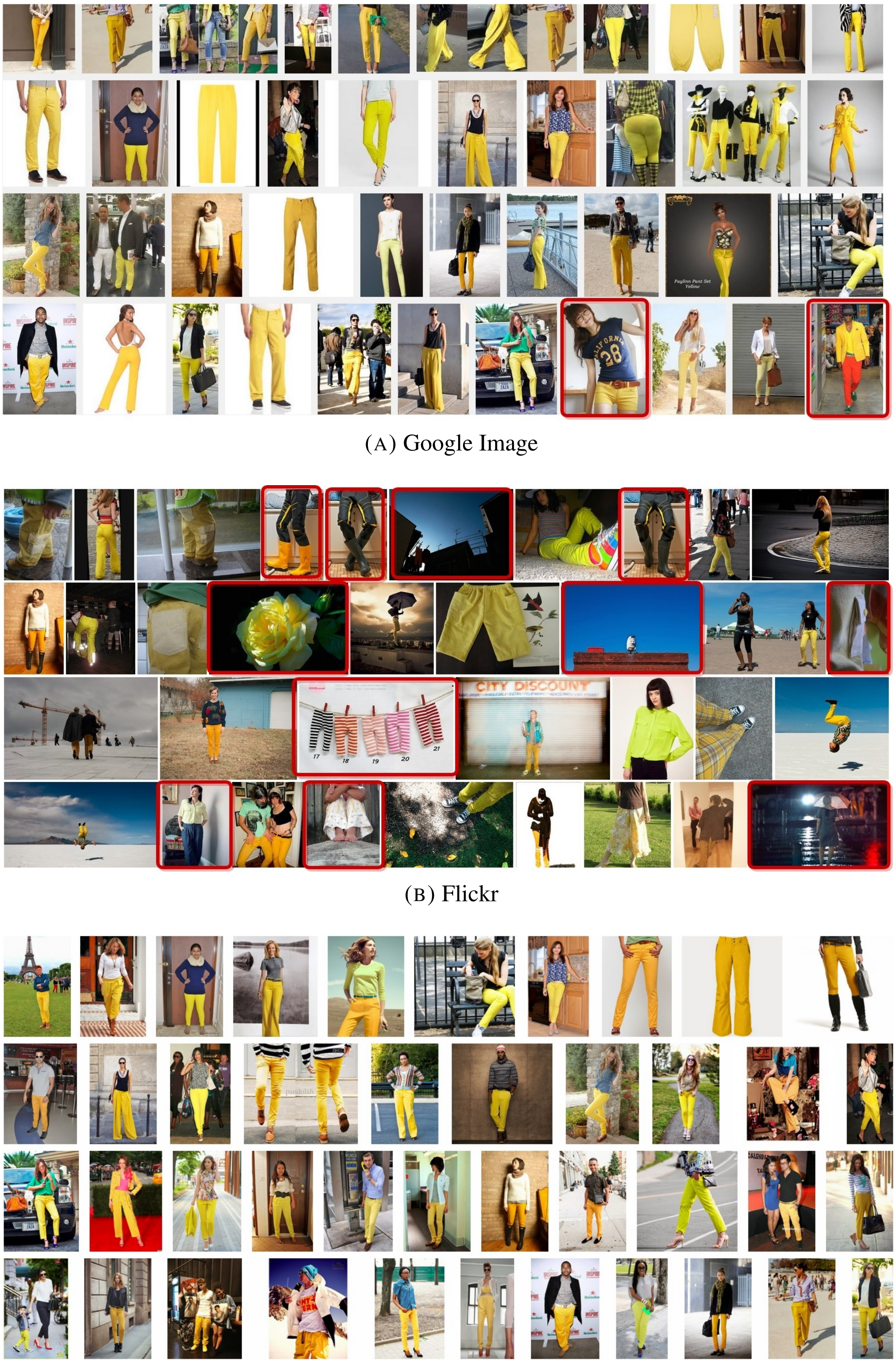


FIGURE 3.8: Tag-based image retrieval framework.

Another benefits of tag-based image retrieval is highly accurate image retrieval results. A search results comparison with Google and Flickr using the keyword “yellow pants” is shown in Figure 3.9. Amongst the top 40 search results, Google and Flickr both return some irrelevant images that have either “yellow” or “pants” text associated with the



(A) Google Image

(B) Flickr

(C) EyeDentifyIt

FIGURE 3.9: Comparison of top 40 search results for the search keyword “yellow pants”. Incorrect results are highlighted in red boxes.

image. In addition, because Flickr users tend to add personal tags which may not reflect the image contents, the retrieval results are certainly lower quality. Further more, tag-based image retrieval outperforms other image search engine when searching multiple objects in one image with "&" sign separating the textual description for each object in the query. For example, users can search "yellow pants & red jacket" to retrieve all images that contains both a pair of yellow pants and a red jacket.

3.6 System Evaluation

EyeDentifyIt is intended to help workers efficiently and consistently generate high quality image tags. The novelties of the system are: an image-click-ads is proposed for the first time as a monetary motivation for workers. Various semi-automated and automated tagging tools utilizing advanced computer vision techniques are designed and implemented for crowdsourcing workers and web publishers. An image hashing scheme is proposed for automatic image tagging when the image has been tagged previously. Therefore, in this section, series of usability study are presented to compare the proposed system to state-of-the-art crowdsourcing system LabelMe, and social computing (image-click-ads) system Thinglink, in terms of worker motivation, workload, data quality and user satisfaction.

3.6.1 Motivation Evaluation

A usability study is presented in this section to support the claim that crowdsourcing workers would want to use EyeDentifyIt to tag images so that they can earn revenues for the work they have done. The more the tag is used, the more revenues from image-click-ads workers can earn. In addition, from the data produced by workers, evidence is presented to show that tags produced by EyeDentifyIt are useful for machine learning algorithms, even in the condition that workers tag images with the primary goal of maximizing image-click-ads revenues.

TABLE 3.3: Conditions in different tasks for motivation evaluation

task	N	conditions	tagging reward (10 images)	image-click-ads reward ($0.1/click for an image)
task 1	21	baseline	$2	0
		image-click-ads	$1	1 click/day: $1 in day 1 (additional) $1 in day 2 (additional)
task 2	21	baseline	$2	0
		image-click-ads	$0.5	2 click/day: $2 in day 1 (additional) $2 in day 2 (additional)
task 3	21	baseline	$2	0
		image-click-ads	$0	3 click/day: $3 in day 1 (additional) $3 in day 2 (additional)

Methodology: A study of three tasks is designed to evaluate how motivated crowdsourcing workers are to tag images comparing baseline condition against image-click-ads (ICA) condition using the same semi-automated tagging tool of EyeDentifyIt. In baseline condition, workers were only rewarded for completing the tagging task. In ICA condition, workers are explained how to tag images as in-image-ads and how they can receive a combination of tagging rewards and ICA rewards depending on the click volume of tagged images. Click volumes are generated by one independent researcher who clicks tagged region of images retrieved by our image search engine. We assume $0.1/click for the ICA reward, which is a common practice in image-click-ads community such as Viglink[5]. Between tasks, baseline condition remains the same while ICA condition varies at different amount of initial task completion rewards and ICA rewards depending on the given visitor volume. For example, in the first task, baseline rewards each worker $2 instantly as the task reward ($0.2/image); ICA condition (assume 1 click/day) rewards $1 as instant task reward, plus additional $1 in one day for two days in a row ($2 in total) as ICA rewards. For ICA condition of different tasks, as click volume increases, ICA compensation accordingly increases while the one-time task reward decreases. Table 3.3 summarizes different payment conditions in different tasks. Participants received a base gratuity of $1 for completing the study. For each task, user's preference on different motivation is explored. Between tasks, the effects

[5] http://www.viglink.com

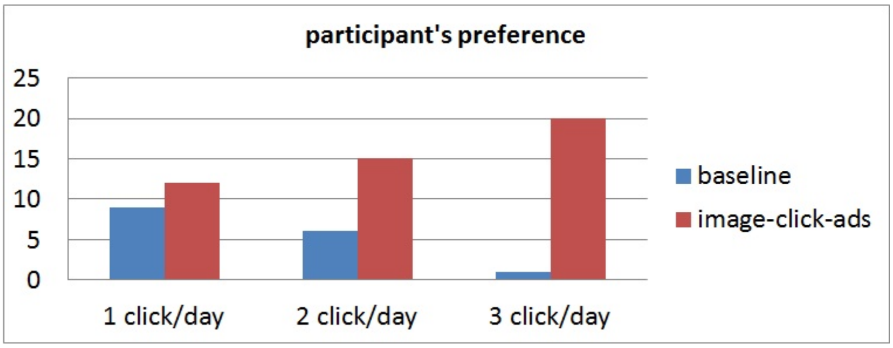


FIGURE 3.10: Distribution of participant's choice for different payment condition.

of different visitor volumes on researcher's cost and worker's preference is studied. Twenty-one participants were recruited for each task. The condition for each task is displayed alternatively between subjects in case there was an ordering effects.

The test data set used in the experiment consists of 685 images from Fashionista data set [4]. All images are posted on a web blog[6], where the JavaScript code generated for the test account ($test\_account@EyeDentifyIt.com$) was installed. Before the experiment, subjects were given an instruction video, showing how to use semi-automated tagging tool. The video also shows in image-click-ads condition, the tagged item is automatically linked to a similar item from ShopStyle[7], which is tracked and monetized in the experiment.

Results: Figure 3.10 illustrates an upward trend with more participants choosing the ICA condition over baseline, at higher visitor volumes. Participants' comments provided some insight as to why they preferred image-click-ads model:

> "earned more gross income without doing extra work as time goes by"

> "image-click-ads has more potential to bring extra earning."

Evidence shows that users input appropriate tags for the images, even in the condition that their primary goal is to maximize their image-click-ads revenues (e.g. goal of web publishers and workers).

[6] http://emtestdataset.blogspot.ca/

[7] http://www.shopstyle.ca/

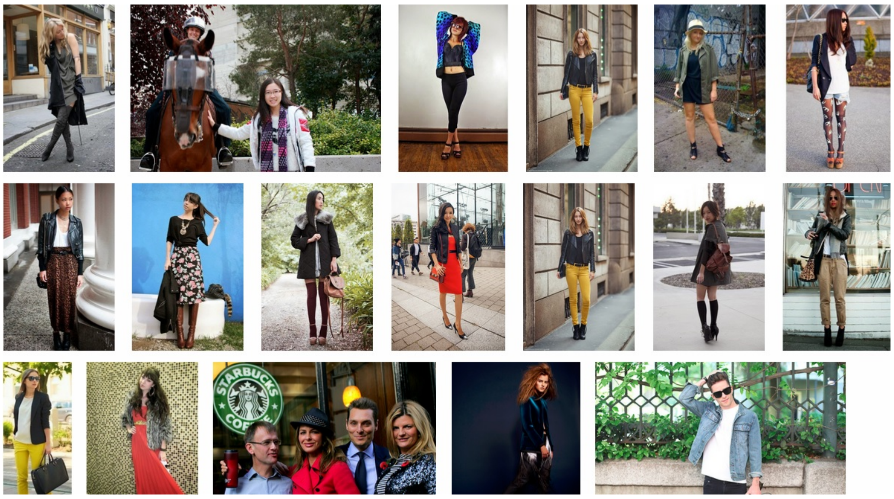


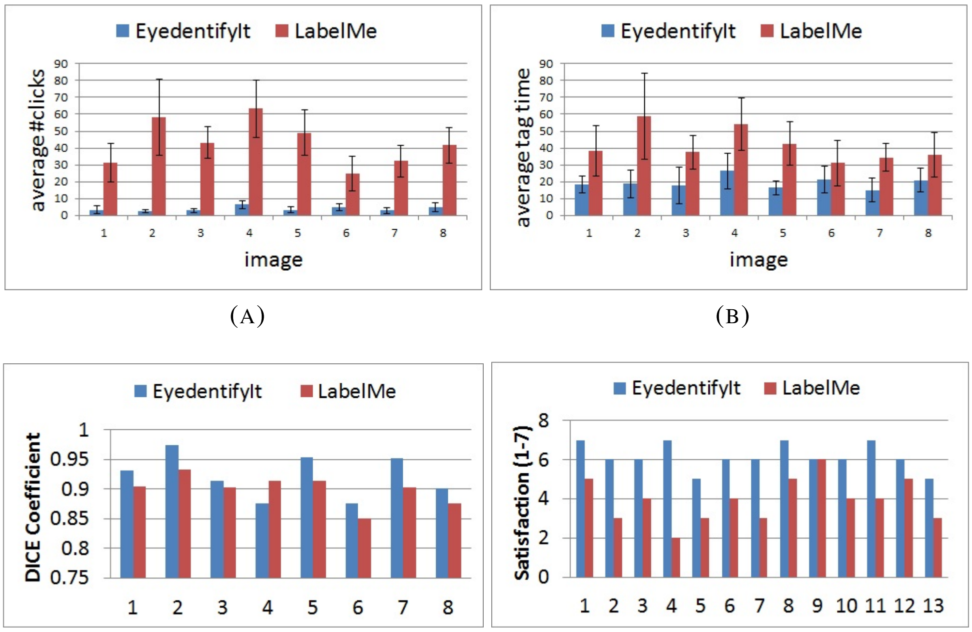
FIGURE 3.11: The first 18 images that had the tag "jacket" associated to them, retrieved by EyeDentifyIt.

An evaluation similar to the one found in [21, 96] was performed. The searching results for all tagged images associated to particular tag are examined. In the study, 10 tags were randomly chosen from the set of all tags collected in the experiment in image-click-ads condition as well as practical users such as web publishers. Figure 3.9 (c) shows the first 40 images having the tag "yellow pants". Figure 3.11 shows the first 18 retrieved images with the tag "jacket". Similar results were obtained for other 8 randomly chosen tags: women, white, coat, top, shirt, blouse, sweater, and shoes. All retrieved images contain contents that truly reflect the test tags. This implies that the search precision of EyeDentifyIt is extremely high.

3.6.2 Vertical Workload Evaluation: Semi-automated Tagging Tool

The semi-automated tagging tool is compared with the manual tagging tool LabelMe used in crowdsourcing platforms, demonstrating it is superior to LabelMe in terms of accuracy, efficiency and user preference.

Methodology: Thirteen subjects (4 females, 9 males) were recruited to use semi-automated tagging tool and LabelMe to tag objects in the selected images as accurately as possible. Eight images from different object categories (sheep, cow, bird, pants, flower, ketch etc) were randomly selected from three different databases with ground truth: Fashionista [4], MSRC[8], and Caltech 101 [79]. Each subject is given a short training demo on the usage of tagging tools. Subjects were allowed to experiment with both tools until they were comfortable to take the test. All interactions with each tool were logged, including the number of clicks, time, and quality of the tags. After completing the task, participants filled out a questionnaire gauging their attitudes toward the tool they used.



(A) (B)

(C) (D)

FIGURE 3.12: Comparison of semi-automated tagging tool and LabelMe. (a) Average number of clicks during tagging process across eight images. (b) Average tagging time across eight images. (c) Tag quality for eight images averaged for subjects. (d) User satisfaction for tag results.

Results: Figure 3.12a-3.12b show the average number of clicks and average tagging time across eight images. Overall, workers using semi-automated tagging tool made 90% less clicks and took 47% less time compared to LabelMe. Figure 3.12c illustrates tag quality comparison for eight images measured using the DICE coefficients [97]. Our semi-automated tagging tool generally tag 5.3% more accurately than LabelMe. In general, users are more satisfied with the tag results of semi-automated tagging tool, as shown in Figure 3.12d.

[8] http://research.microsoft.com/en-us/projects/ObjectClassRecognition/

3.6.3 Vertical Workload Evaluation: Automated Tagging Tool

The automated tagging tool is compared with the image tagging tool Thinglink, which is commonly used in commercial image-click-ads (social computing) platforms. It is demonstrated that our automated tagging tool is superior to Thinglink in terms of efficiency and user satisfactory.

Methodology: Thirteen web publishers were recruited to use automated tagging tool and Thinglink to tag a set of eight images. Images were randomly selected from Thinglink website, amongst which the first four images containing objects that can be automatically recognized by our automated tagging tool. Tagging time and user feedback were recorded. None had used Thinglink or our tool before. Nine have image tagging experiences using Flickr, Facebook etc.

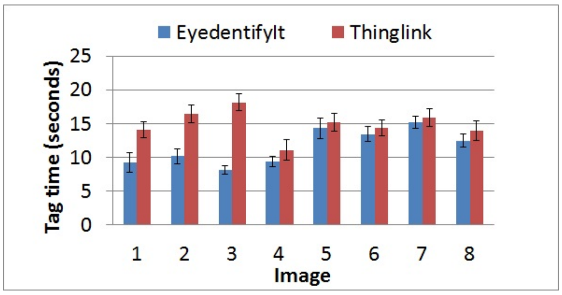
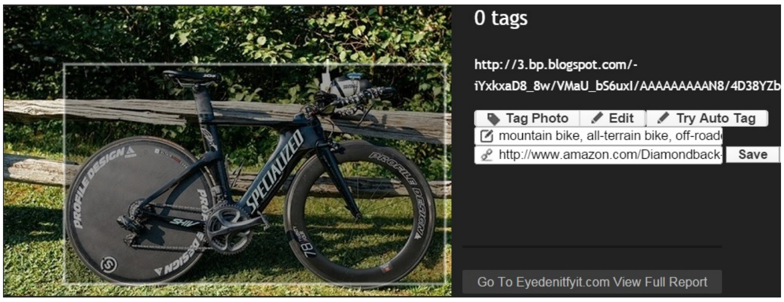


FIGURE 3.13: Comparison of tagging time between EyeDentifyIt and Thinglink.


0 tags

http://3.bp.blogspot.com/-iYxkxaD8_8w/VMaU_bS6uxI/AAAAAAAAAN8/4D38YZb

mountain bike, all-terrain bike, off-road

http://www.amazon.com/Diamondback

FIGURE 3.14: The automatic tagging tool fills the tag description and tag link automatically.

Results: Subjects generally reported our tool is “easier and way more efficient once one understood how it works”. Overall, they reported the preference for automated tagging tool over Thinglink. For the objects that can be automatically recognized by our tool, the tagging time was 38.2% less than Thinglink. Each recognized object has $1\sim4$ descriptive labels, 1.3% more labels than tags on average for the testing images. For example, as shown in Figure 3.14, the object is automatically recognized for three tag descriptions: “mountain bike”,“all-terrain bike”, and “off-roader”. More tag descriptions can potentially increase the image-click-ads revenue as it has higher exposure probability during the image retrieval, as discussed in Section 3.5. For images without successful auto recognition, automated tagging tool takes about the same time as Thinglink, with a slight tagging time improvement of 6.5%.

3.6.4 Horizontal Workload Evaluation

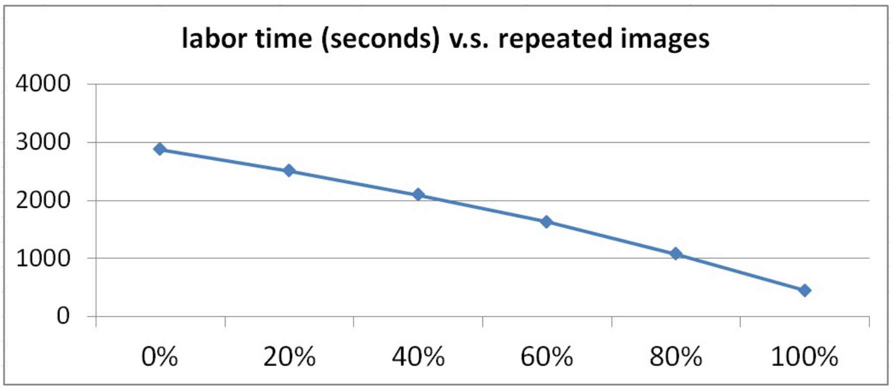


FIGURE 3.15: Tagging time for different duplication percentage.

Methodology: A user study was conducted to evaluate the efficiency of the system, particularly in cases that part of the image data set has been tagged by someone else. In the controlled study, two researchers were recruited to manually tag 90 images in six different tasks. In each task, the researcher needs to tag three fashion items with color label for each image, 15 images in total. Each task is different in the percentage of images that are identical in content (but different URL)[9] to existing tagged images in the system, which are tagged with fashion items only (with no color label). The percentage of identical images is increased by 20% for each task. Such change of exiting tags is common during the creation of tagged data sets. For example, colorful fashion data

[9] This ensures the existing tags for identical images can only be retrieved through the image hashing

set[10] from Liu *et al* [98] requires color label associated with fashion item information, such as “red pants” instead of “pants”.

Results: Time for finishing each task was recorded, and averaged for two subjects. As shown in Figure 3.15, as the percentage of identical tagged images increases, the tagging time deceases roughly in linear. On average, tagging a new image takes about 191.5 seconds, and editing exiting tags takes about 30.1 seconds per image, which is a 84.3% improvement in time. When the subject chooses to keep exiting tags, it only took about 10.1 seconds for each image, which is a 94.7% improvement in time. Another observation is that as more images get tagged, the tagging and editing time for new images also decreases, probably due to the familiarity of the tagger with the tool.

[10] https://sites.google.com/site/fashionparsing/dataset

Chapter 4

Automatic Tagging of Fashion Images

During the formative studies presented in the Chapter 2, we observed that social web users are free workers who prefer minimum workload (automated tagging tools) when it comes to tagging images, which lowers tagged data quality. Crowdsourcing workers are willing to take more workload (semi-automated tagging tools) to tag images at higher quality, because of initial task rewards from the task assigner, which although has been reduced in SMC model comparing to crowdsourcing. Therefore, if we can reduce the workload of crowdsourcing workers to a level of social web users while keeping tagged data quality, we can further reduce the cost for a task assigner. A method for automatically segmenting and tagging regions will not only reduce workload, but also provide high tagging quality.

In this chapter, we will present an efficient method for automatically parsing fashion photos given a list of tag names, which resolves many common problems in state-of-the-art fashion parsing methods, e.g. occlusions, background spills and inaccurate initial pose estimates. Also, the efficient computation of the proposed parsing method allows it to be integrated in the image tagging system EyedentifyIt.

Our automated image parsing method focuses on fashion images, because they are playing a major role in fashion market, e-commerce, and image-click-ads platforms. There is also a group of highly motivated social web users ready to tag fashion images. For example, many fashionistas share their fashion photos to some website[1] and usually provide some information related to the garment items that appear in each photo. There is a great need for tagging these images for in-image-ads purpose. However, this does not limit our image tagging system to be integrated with other automatic image parsing methods.

In this chapter, we first review existing methods of fashion image parsing. Then we describe the training data set we used to test and train our algorithm. Section 4.4 describes our new fashion image parsing method that help users to tag clothing items more efficiently and accurately. Section 4.5 presents the evaluation results comparing our method to state-of-the-art methods.

4.1 Background and Related Work

4.1.1 Image Parsing

Image parsing, also called semantic segmentation, refers to the task of segmenting an image into semantically meaningful regions where each region is labeled with a specific object class [99]. The existing approaches tackle the problem from various aspects: elementary regions (e.g. globle, regional, local), features to describe regions (e.g. textons [100], features learned from large-scale deep Convolutional Neural Networks [101]), spatial relationship modelling (e.g. shape and pose cues [102]), incorporation of context [3], and different optimization techniques (e.g. back propagation, graph cut) etc. The most successful approaches typically use Markov Random Fields (MRFs) [103] or its variant Conditional Random Fields (CRFs) [3, 82]. A MRF typically formulates a probabilistic generative framework modelling the joint probability of an image and its corresponding labels [104, 105]. It usually incorporates local relationships between neighbouring nodes. This allows the model to locally smooth the assigned labels, based on local regularities. In contrast, CRF model the conditional probability of labels given an image, which will likely depend on structures at different level of granularity in the image. This conditional probability model can depend on arbitrary non-independent

[1] www.chitopia.com

characteristics of the observation. CRF can employ a feature function that encodes a particular pattern within a subset of label variables. For example, as shown in Figure 4.1, He *et al* [3] proposed to learn label features at regional (a pattern of ground pixels above water pixels) as well as global (rhino/hippo in the water with sky above the horizon) based on a set of labeled images.

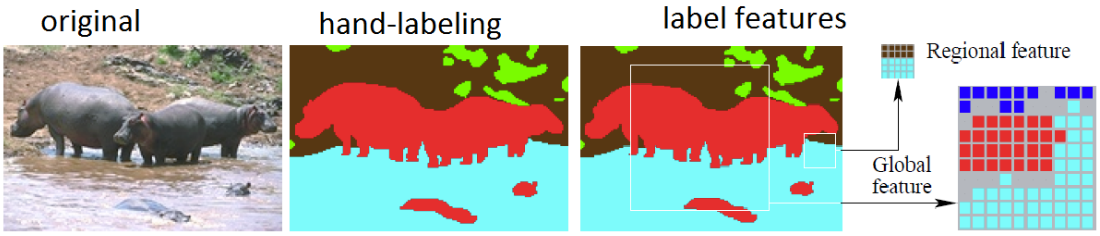


FIGURE 4.1: An example of CRF modeling [3]. From left to right: original image, ground truth labeling, and example label features.

We particularly consider the fashion photo parsing problem in this chapter, which is a subcategory of image parsing. We use a related approach, MRF based labeling used in the field of image parsing. The key insight is that conditional probability model learned from a training data set may not be as reliable as local features of the query image, especially when the learned model cannot truly reflect the statistical structure of the query image. This is particularly true for fashion images, because there are various combinations of large number of garment items and great variations in their appearance, layering, and occlusion etc. For example, Yamaguchi *et al* [4] employed a CRF model to incorporate a prior distribution over the pairwise co-occurrence of clothing labels in neighbouring regions, and the probability of neighbouring pairs having the same label given their features, learned from the training data set. Such prior distribution will certainly fail if the queried image contains features that were not seen in the training data set (e.g. Fashionista [4] used in the training and evaluation of Yamaguchi *et al* [4] does not include images of yellow pants). In contrast, we build models to incorporate many characteristics of fashion images. We employ the background prior to model clothing items which are located in salient regions of an image. An occlusion prior is proposed to model the occlusion due to clothing layering. We apply re-weighted pairwise term in the MRF model to justify weak responses to infrequent labels (e.g. necklace) assigned to small regions. To our best knowledge, it is the first time that background prior, occlusion prior and re-weighted pairwise term are applied to MRF inference model. As will be demonstrated in the evaluation section, our MRF method achieves better performances than the comparable CRF method.

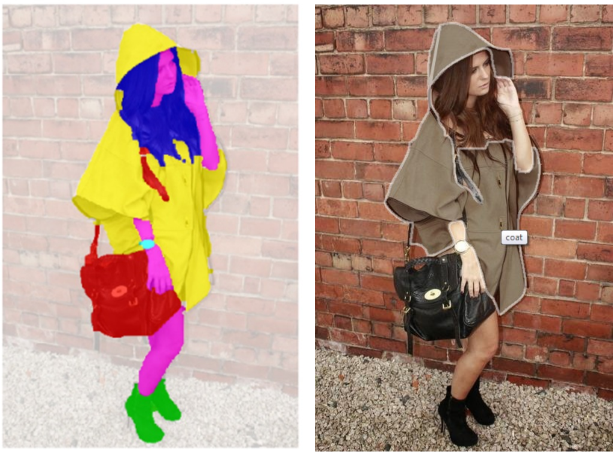
4.1.2 Clothing Recognition

Fashion image parsing is a relatively new research area in both computer vision [106] and computer graphics [107]. It has received great attentions recently because of its importance in large fashion market and e-commerce applications, such as clothes recommendation and retrieval. Fashion parsing is an extremely challenging problem because of the large number of possible garment items and possible variations in appearance, layering, occlusion, and combination. The characteristics of this particular type of images such as human pose detection, clothing layering, and occlusion make the parsing problem different from parsing natural scene images.

Many work have focused on some specific aspects of cloth recognition, e.g. predicting attributes of clothing [108–110]. There are also work on cloth recommendation [111], and identifying social identity through clothing [112, 113]. However, none of their work addresses the clothing parsing problem.

The first work in clothing parsing was done by Hasan *et al* [114]. They proposed to incorporate a shape prior model to a Markov Random Field (MRF) formulation. However, they only considered 4 categories: shirt, jacket, tie, face, and skin. Liu *et al* [98] resolved weakly supervised parsing problem, which means their training data are image-level color-category tags rather than pixel-level labels. Dong *et al* [115] used parselets as the building blocks for training and parsing process. A parselet is a group of semantic image segments obtained from a low-level over segmentation algorithm. Another category of approach is based on data driven approach. Yamaguchi *et al* [116] proposed to predict the parsing of a queried image by retrieving similar outfits from the parsing of a known database, building local models from the retrieved clothing items, and transferring the inferred clothing model from the retrieved samples to the query image. However, such method is highly limited to a pre-defined data set and cannot be applied in general.

Our work is similar to Yamaguchi *et al* [4], as we use similar inputs and produce comparable outputs. For the input, we both use manually tagged images in a training phase for the same number of classes. In the query phase, a user needs to provide a list of tags to be parsed along with the query image. For the output, we both produce segmented regions with each region associated to a tag. However, our approach is quite different from Yamaguchi *et al* [4] in many ways: We do not use superpixel level tagged data during the training phase but using tagged images provided by EyeDentifyIt. We also employ a MRF framework incorporating background prior, occlusion prior, and re-weighted pairwise term rather than CRF model. Yamaguchi *et al* [4] also do not consider clothing parsing for real-time applications, e.g. online tagging platform. Therefore their parsing method is not efficient on memory usage and processing time.



(A) (B)

FIGURE 4.2: Data set imported in EyedentifyIt. (a) Colour coded label map of the labeled image after importing to EyeDentifyIt format. (b) Visualization of the label "coat" on the web enabled by EyeDentifyIt.

4.2 Overview of The Proposed Method

4.3 Fashionista Data Set in EyeDentifyIt

We imported the Fashionista data set [4] into a format that is compatible with EyeDentifyIt as the training images. Figure 4.2 shows an example image of the imported data set. The original Fashionista was tagged using workers from Amazon Mechanical Turk who created ground truth clothing tags on pre-segmented superpixel[2] regions. After importing into EyeDentifyIt, superpixels with the same label will be merged into one region for future feature extraction and training etc. The ground truth data set contains 685 images with good visibility of the full body and covering 53 different clothing tags, e.g. dress, bag, blouse, *hair*, *skin*, and *null (background)*.

Other than benefits for better interactivity, visualization, and ease of modification that EyeDentifyIt provides, there are three major advantages for importing Fashionista into EyeDentifyIt. First, it is easier to add more tagging data to increase the size of the training data set. Almost all existing tagging tools (e.g. LabelMe) require users to tag complete region of the fashion item rather than superpixel patches. The imported data set allows the labeled images to be accommodated by other existing tagging tools. Second, training on tagged regions is faster than superpixel patches because the number of tagged region is significantly less than the superpixel patches. Third, models learned from tagged regions are more accurate than superpixel patches. Some features, such as texture patterns (common on fashion items), cannot be extracted from isolated superpixel patches. Take Convolutional Neural Network (CNN) model [89] for example, the features are trained from images tagged with tight bounding boxes, e.g. ILSVRC1K [90] data set.

4.4 Proposed Method

In this section, we describe the proposed approach for parsing fashion images, including the formal definition of the parsing problem. For a query image, we start from a successful pose estimation and superpixel segmentation. We then formulate the parsing problem using MRF algorithm that can predict tags for superpixel patches. The approach utilizes a background prior, occlusion prior, and a re-weighted pairwise contrast term, which improve parsing results on both processing accuracy and efficiency.

[2] Superpixel refers to segmenting an image into sets of compact and nearly uniform pixels grouped based on color and texture etc similarities

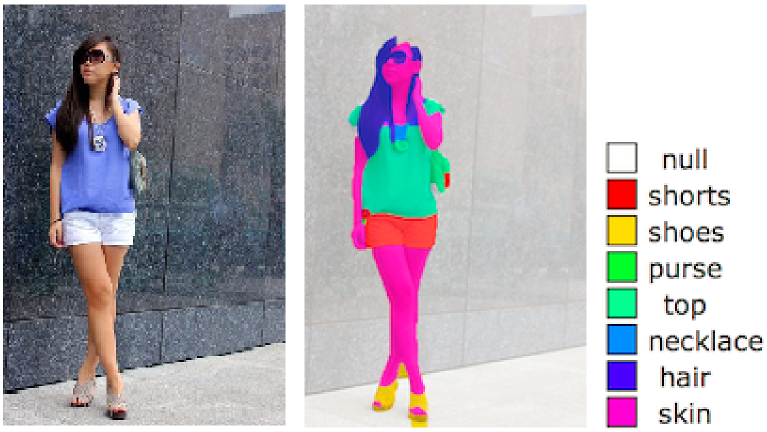


FIGURE 4.3: An example of semantic segmentation for fashion images [4].

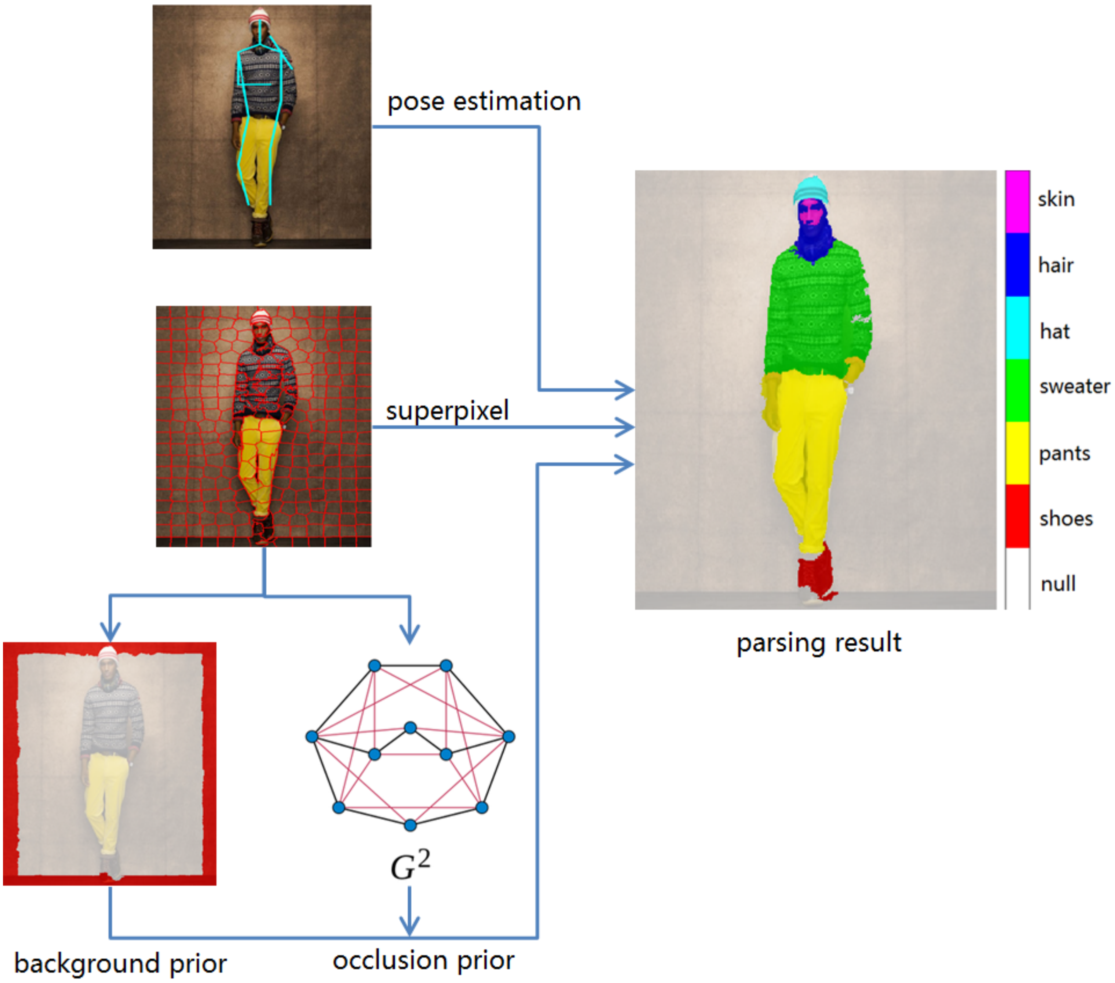


FIGURE 4.4: Proposed automatic clothing segmentation pipeline.

4.4.1 Problem Formulation

The image parsing problem can be mathematically defined as: given an image I and a pre-defined tag set L, the goal is to find a proper assignment L^* between the tags and the image sites (pixel, superpixel or block) that maximizes the conditional probability $P(L|I)$

$$L^* = \arg\max_L P(L|I). \tag{4.1}$$

The clothing parsing problem can be formulated as a pixel level tagging problem. One example of clothing parsing is shown in Figure 4.3. The goal is to assign a clothing tag (e.g. top, pants), skin, hair, or null (background) to each pixel. Because tagging each pixel is not very computationally efficient, we simplify the problem by grouping uniform pixels to the same superpixel region, and reduce the problem to a graph labeling process over a set of superpixels. Let $I = \{s_i\}_{i \in S}$ denote a fashion image showing a person, where s_i is the data from the i^{th} patch of the superpixel set S. We formulate the problem as a graph model that finds the solution L^*, minimizing an energy function defined as:

$$\begin{aligned} L^* &= \arg\min_L (E(L)) \\ &= \arg\min_L (E_{data}(L) + E_{smooth}(L)). \end{aligned} \tag{4.2}$$

The unary term accounts for the cost to assign a tag to a superpixel patch according to its feature. One important feature for cloth parsing is the human pose configuration, denoted by $X = \{x_p\}$, where x_p is a set of image coordinates for the body joint p, such as head and neck. The pairwise term accounts for the cost to assign a pair of tags to neighbouring patches, which incorporates region contrasts. They can be represented by:

$$E_{data}(L) = \sum_{i \in S} \Psi_1(l_i|X, I) \tag{4.3}$$

and

$$E_{smooth}(L) = \sum_{(i,j) \in V} \Psi_2(l_i, l_j|X, I), \tag{4.4}$$

where Ψ_1 and Ψ_2 are unary and pairwise term respectively, and V is the set of neighbouring patches. Figure 4.4 shows the processing pipeline defined as:

1. Estimate pose configuration $\{x_p\}$ from image I
2. Obtain superpixel $\{s_i\}$ from image I
3. Obtain background prior and occlusion prior from $\{s_i\}$
4. Predict label assignments L^* using $E(L)$.

In the following subsections, we will describe each step in detail and formally define the MRF model and its optimization.

4.4.2 Pose Estimation and Superpixels

Two common pre-processing steps when performing fashion image parsing is pose estimation and superpixel segmentation. Figure 4.4 shows an example of the pose estimation and the SLIC segmentation from a given image computed in our method. Most of fashion images contain one single person with relatively simple pose, and the person's clothing appearance is in general highly correlated to his/her pose. Under such assumption, human skeleton provides important clue to fashion tag prediction. For example, belt can only appear near the waist. As in other similar work [4, 98, 117, 118], we use state-of-the-art pose estimation algorithm described in [119] to compute the locations of 14 joints, such as head, neck, and left/right knees, represented as $X = \{x_p\}$ and computed as in [119] by:

$$X = \arg\max_X P(X|I). \tag{4.5}$$

Both Yamaguchi *et al* [4] and our work use a superpixel segmentation in the preprocessing. We both assume each patch contains similar pixels and all pixels in one patch belong to the same category, so that the superpixel can be used as the building block in our processing pipeline, which highly reduces the computational cost. Yamaguchi *et al* [4] used a hierarchical segmentation algorithm [5], which yields a more intelligent segmentation (merging similar regions in a hierarchical structure) by sacrificing the computational cost. In contrast, we obtain more regularly gridded homogeneous regions $\{s_i\}$ using a naive but much more efficient superpixel segmentation algorithm

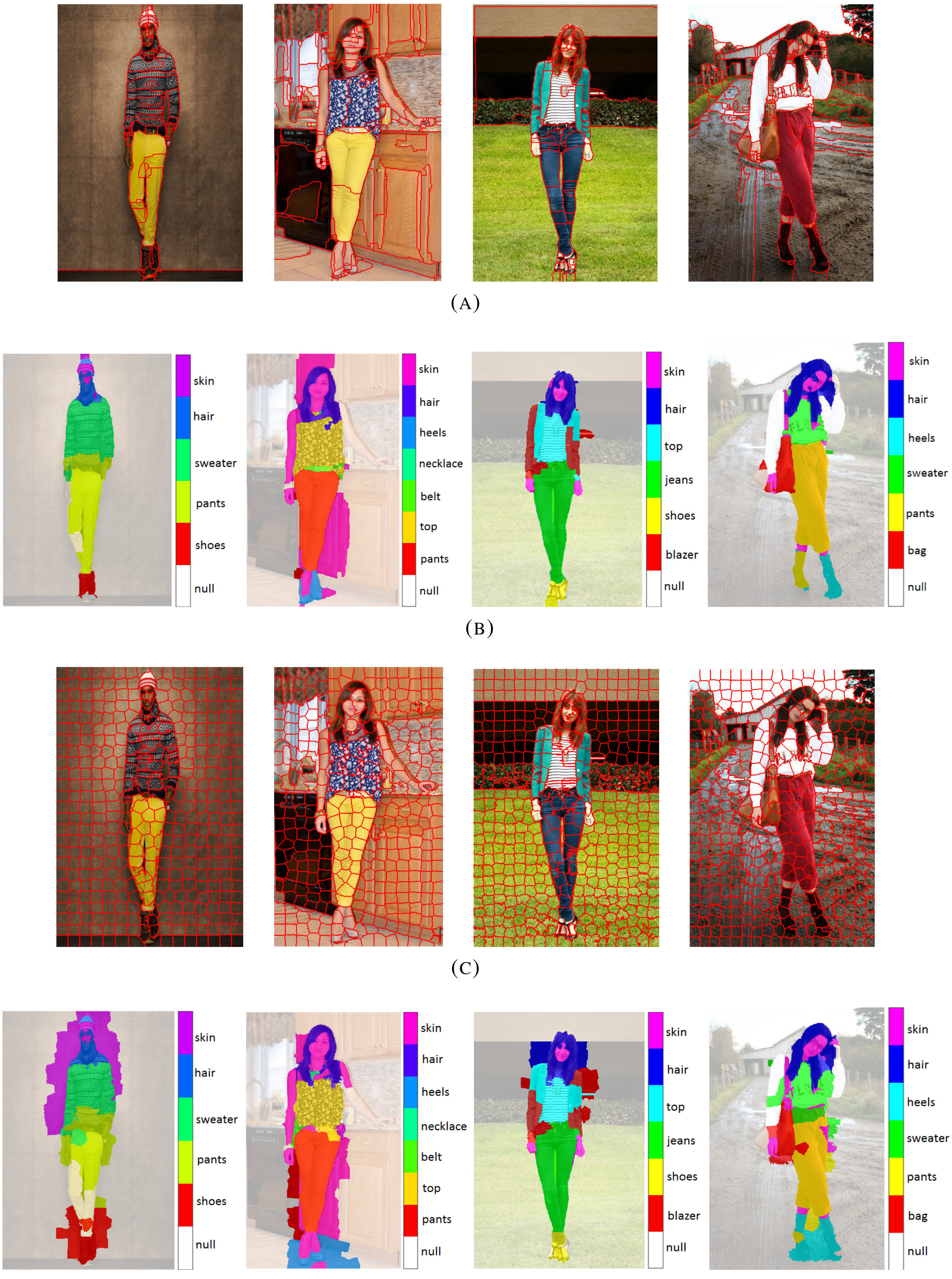


FIGURE 4.5: Illustration of clothing parsing results by Yamaguchi *et al* [4] using different superpixel segmentation methods: (top row) hierarchical superpixel segmentations [5], (second row) the parsing results using superpixel segmentation in top row, (thrid row) SLIC superpixel segmentations [6], and (last row) the parsing results using superpixel segmentation in third row.

named SLIC [6]. As shown in Figure 4.5, SLIC superpixel regions (regular segmentation) are less intelligent comparing to the hierarchical superpixels [5] used in Yamaguchi *et al* [4], which poses more challenges for our follow-up processing steps. When we replace the hierarchical segmentation algorithm [5] by the SLIC algorithm [6] in the pre-processing step of Yamaguchi *et al* [4], the same model yields much poorer parsing results, as shown in Figure 4.5b and Figure 4.5d. This experiment in the pre-precessing step also raises a question: is the CRF model really better than the MRF model? Is it the CRF model itself or the hierarchical segmentation that helps the parsing preserve good region contrasts? Why does the model lost the ability of preserving region contrasts after replacing the hierarchical segmentation by a more naive superpixel segmentation? Such observation inspires us to pursue the MRF framework, which relies on the local features of the query image to preserve good region contrasts. On the other hand, SLIC segmentation [6] yielded 93.3% performance increase comparing to hierarchical segmentation algorithm [5] for images in Figure 4.5. Such efficient computation allows us to apply the parsing method in many real-time applications, e.g. image tagging platform EyeDentifyIt.

4.4.3 Background Prior (BP)

As opposed to previous work, we explore how a background prior can be used to assist fashion parsing. Background prior was first proposed by Wei *et al* [120] for tackling object level saliency in a detection problem. It is based on two observations: salient objects do not touch image boundary (namely boundary prior), and backgrounds are continuous and homogeneous (namely connectivity prior). It is a basic rule of photographic composition that most photographers will not crop salient objects along the view frame. We indiscriminately refer them as the background prior in this thesis. Because in fashion images, garments are usually attached to the human body which is a silent object, one can apply the same rule to fashion images to distinguish foreground from background.

As proposed by Wei *et al* [120], we also assume all the boundary patches are background. As shown in Figure 4.4, the superpixel patches around the image borders are assigned as background nodes, represented as:

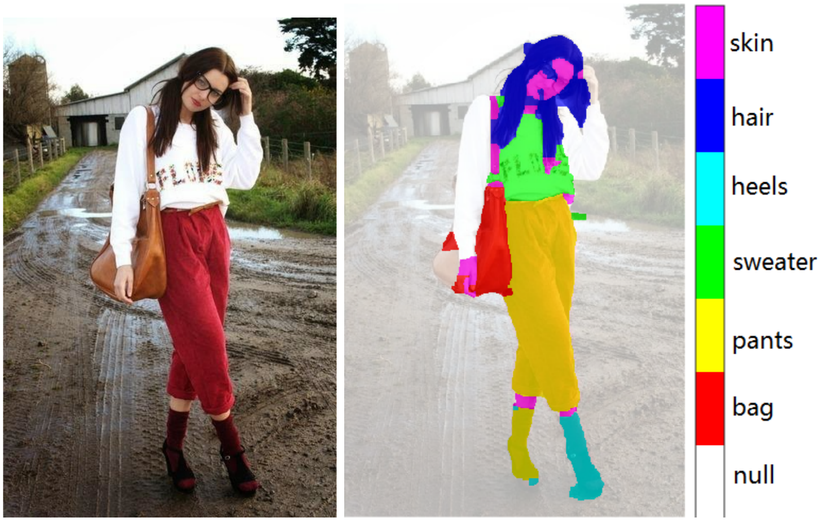
$$\Psi_1(l_i = null|X, I) = 0. \tag{4.6}$$

Applying background prior has two major advantages. First, it further reduces the computational complexity for patches on the image borders. Second, it provides a prior to group homogeneous background regions together in an energy optimization computation used to perform the tagging of the fashion items.

4.4.4 Occlusion Prior (OP)

In fashion images, because of a large variety of human poses, clothing layering and configuration, very often clothing items (as well as background) are occluded. In such cases, when the object is partially occluded and separated into separate pieces, the separated pieces may be assigned different tags even when they have the similar feature, e.g. color. One such example is shown in Figure 4.6. Because the sweater is partially occluded by the bag belt and hairs on the shoulders, the sweater on the left arm and the right arm is visually isolated from the body piece. Therefore the computed sweater tag by Yamaguchi *et al* [4] is not completed (predicted as "null" for the arm parts) in this example. This is a common problem in most of graphical (CRF and MRF) models. Because the pairwise term only considers neighboring patches, clothing patches that are occluded (separated) by other objects will not be connected by an edge in the graphical model.

We tackle this problem by considering neighbours of neighbourhood (NON) relationship between nodes in MRF as neighbourhood as well. Intuitively, adding edges between NON nodes would help smoothing tag assignment for object regions (nodes) that are occluded (separated) by other objects. The idea is that NON patches (potentially occluded object) should have similar tags, if their features are similar.



(A) original image

(B) parsing result by Yamaguchi *et al* [4]

FIGURE 4.6: Illustration of the occlusion problem.

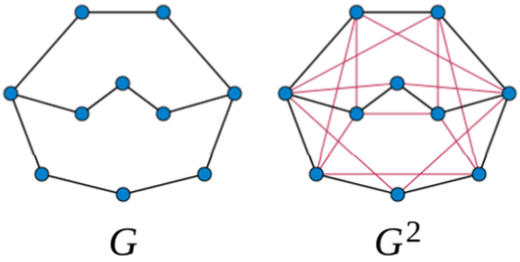


FIGURE 4.7: Square of graph [7]: NON edges in red in G^2 is computed from G.

Suppose the neighbourhood set is V, our goal is to find a new neighbourhood set V^* that includes NON, represented as:

$$V^* = V + NON. \tag{4.7}$$

We use the graph power to compute NON. The kth power G^k of an undirected graph G is defined as another graph that has the same set of vertices but two vertices are considered adjacent when their distance in G is at most k. Figure 4.7 shows an example of G^2 from Wikipedia. In our case $k = 2$, because we look for NON relationship between nodes. It is computed by building an adjacency matrix A for the graph, then the non zero entries of A^k give the adjacency matrix of the kth power of the graph.

4.4.5 Energy Function and Global Optimization

A) Feature Vector Before computing the pixel tags L, a feature vector of five elements is extracted from each pixel as their global feature representation. These features are:

- RGB color [$m \times n \times 3$]: red, green, blue color channel of an image, each channel with a value in the range $[0, 255]$;
- CIELAB color [$m \times n \times 3$]: a color model adopted by CIE [121] in 1976 that better describes uniform color spacing in their values, with dimension L for lightness and a and b for the color-opponent dimensions;
- Gabor feature [$m \times n \times 4$]: a feature filter used for edge detection;
- absolute 2D coordinates [$m \times n \times 2$]: absolute 2D coordinates of each pixel;
- relative 2D coordinates [$m \times n \times 28$]: 2D coordinates of each pixel relative to each body joint location x_p;

where $[m, n]$ is the size of image I. Each feature is normalized in a 10 bins histogram independently and then concatenated to form the final feature vector of [$m \times n \times 400$], which is aggregated by superpixel patches, represented as $\phi(s_i, X)$.

B) Unary Term: Classifier Potential With a feature vector for each superpixel patch $\phi(s_i, X)$ and parameters of the trained model θ (training part refers to Section 4.4.6), we model the unary term using the probability of a tag assignment for each super patch:

$$\Psi_1(l_i|X, I) = -\lambda_1 \ln P(l_i|\phi(s_i, X), \theta). \tag{4.8}$$

C) Pairwise Term: Re-Weighted (RW) Spatial Smoothness The pairwise term favors piecewise constant tag map. The idea is that neighboring patches should have similar tags, especially if their colors are close. Such pairwise prior can reduce the effects of imperfect pose detection results. The pairwise term $\Psi_2(l_i, l_j|X, I)$ encodes neighboring nodes affinity through edge weights such that nodes connected by edges with high-weight are considered to be strongly connected and edges with low-weights are disconnected nodes. The patch affinity is measured using the Euclidean distance between mean colors (in CIELAB color space) of two patches s_i and s_j, represented as:

$$w = max(\|I_i - I_j\|, \gamma), \tag{4.9}$$

where γ is a clipping distance, in order to prevent division by zero in pairwise term computation (Equation 4.10) and to limit the overflow of the total energy $E(L)$ (larger than MAX_ENERGY) in Equation (4.2). The pairwise term in our model is defined to measure re-weighted spatial smoothness, represented as:

$$\begin{aligned}\Psi_2(l_i, l_j|X, I) = \lambda_2 \times & \frac{max\Big(\Psi_1(l_i|X, I)\Big) + max\Big(\Psi_1(l_j|X, I)\Big)}{2 \times w} \\ & \times \exp(-\beta \times w),\end{aligned} \tag{4.10}$$

where $\frac{1}{2\beta}$ is the average square distance between color vectors for adjacent patches in an image. The pairwise term in Equation 4.10 assigns the edge weight between two nodes using intensity difference re-weighted by the data term of two nodes. Such re-weighted smoothing term is a result of experimental analysis. In our experiment, re-weighted term can avoid over smoothing infrequent (small) labels, such as necklace. The effect of reweighing is demonstrated in the experimental section. In comparison, a common MRF without re-weighted pairwise term is defined as:

$$\Psi_2(l_i, l_j|X, I) = \lambda_2 \times \exp(-\beta \times w). \tag{4.11}$$

4.4.6 Training and Inference

The trained model θ in Equation (4.8) mainly includes the probability distribution $P(l_i|\phi(s_i, X))$ using logistic regression with L2 regularization (liblinear library [122]). Because we use MRF model, there is no need to compute the pairwise model that is the probability of neighbouring pairs having the same tag, represented as $P(l_i = l_j|\phi(s_i, s_j, X))$. Therefore, our training time is reduced significantly comparing to the CRF model. We experimentally chose γ and β, and find the best the values for λ_1 and λ_2 by maximizing the cross validation of pixel accuracy in the training data. In our experiment, typical values are $\gamma \in [1,5]$, $\beta \in [0.1, 0.3]$, $\lambda_1 \in [19, 23]$ and $\lambda_2 \in [1,3]$. The main computational cost of our parsing model comes from the MRF inference step. We use alpha-expansion implemented using the gco-v3.0 library [123] to solve the multi-label optimization problem. The computational complexity is $O(|S||L|)$, where $|S|$ is the number of superpixel patches, and $|L|$ is the number of clothing labels.

4.5 Experiments

We carried out different sets of experimental evaluation to analyze quantitatively and qualitatively the performance of our algorithm. We mainly compared our results with the state-of-the-art clothing parsing method by Yamaguichi *et al* [4]. All measurements use 10-fold cross validation.

4.5.1 Quantitative Evaluation

A) Experimental Setting We evaluate the performance of our method using images in Fashionista data set against the algorithm described in Yamaguichi *et al* [4]. Two criteria were used in our evaluation, average pixel accuracy (ACC) and Mean Average Garment Recall (MAGR), defined respectively as:

$$ACC = \sum_{i=1}^{N} \left(\frac{I_i(\text{\# of Pixels of True Pos Labels})}{I_i(\text{\# of Total Pixels})} \right) /N \tag{4.12}$$

and

$$MARG = \sum_{i=1}^{N} \left(\frac{I_i(\text{\# of True Pos Labels})}{I_i(\text{\# True Pos Labels+ \# of False Neg Labels})} \right) /N, \tag{4.13}$$

where N is the number of images in the data set.

TABLE 4.1: Labeling accuracy and recall rate for using MRF only, MRF with re-weighted pairwise term (RW), re-weighted MRF with background prior (RW+BP), and re-weighted MRF with both background prior and occlusion prior (RW+BP+OP).

	MRF	**RW**	**RW+BP**	**RW+BP+OP**
Pixel ACC	87.3%	88.8%	89.7%	**90.5%**
MAGR	61.5%	**63.0%**	62.8%	61.4%

TABLE 4.2: Clothing parsing performance comparison: results are shown for our model, CRF method by Yamaguichi *et al* [4] and a baseline labeling.

Method	**Pixel acc**	**MAGR**	**Training Time**	**Processing Time**
ours	90.5%	63.0%	631.8 sec	5.2 sec
Yamaguichi *et al* [4]	85.1%	57.2%	4546.7 sec	81.5 sec
baseline	77.6%	12.8%	N/A	N/A

As Yamaguichi *et al* [4] method, we experimentally chose the best model parameters that maximize the pixel ACC in all our experiments. No ground truth pose information is used in all computations.

B) Different Version Comparison We compared four different versions of our method and summarised the results in Table 4.1, including MRF, re-weighted MRF (RW), re-weighted MRF with background prior (BP+OP), and re-weighted MRF with both background prior and occlusion prior (RW+BP+OP). RW outperforms MRF on MAGR mainly because the re-weighted pairwise term can avoid over smoothing infrequent tags (small region). The pixel ACC steadily improves with more prior information, with RW+BP+OP reaching the best pixel accuracy of 90.5%. However, BP and OP also cause slight drop on MAGR because of potential over smoothing (loss of labels) for infrequent labels. BP increases the chance of a node being assigned as background, and OP increases the chance of a node being assigned the same label as its NON. More visual parsing results can be found in the Section 4.5.2.

C) Overall Performance We compared the performance of our method with CRF model by Yamaguichi *et al* [4]. Table 4.2 summarizes the results of the performance comparison. The baseline method naively predicts all regions to be background, this results in 77.6% pixel accuracy and 12.8% MAGR. The CRF model by Yamaguichi *et al* [4] obtained 85.1% pixel ACC and 57.2% MAGR respectively. In comparison, our algorithm obtained 90.5% pixel ACC and 63.0% MAGR, with a 5.4% gain on pixel ACC and a 5.8% gain on MAGR over the CRF model by Yamaguichi *et al* [4]. In addition, our algorithm has significant improvements on the training time and processing time. Since we use the MRF model, we only need to train the global model for the unary potential. No pairwise model needs to be trained over pairwise clothing tags and features of neighbouring pairs as in CRF model. This gives us a 86.1% improvement on the training time. Compared to the expensive hierarchical segmentation algorithm [5] used in the CRF model, the robustness of our model allows us to use a more naive segmentation algorithm (SLIC), which results a 93.6% improvement on the processing time. Efficient processing makes our method more applicable to real-time applications such as web-based image tagging.

4.5.2 Qualitative Evaluation

In this section, we first qualitatively compare the parsing results between the CRF model by Yamaguichi *et al* [4] and our algorithm. Figure 4.8 shows the parsing results of eight test cases: four images (from Figure 4.8a to Figure 4.8d) randomly downloaded from the web and four images (from Figure 4.8e to Figure 4.8h) from Fashionista data set. From the results, one can see that our method performs better on preserving tag smoothness (in line with region contrasts) as well as retaining infrequent region tags, e.g. the hat in Figure 4.8c and the shoes in Figure 4.8g. Figure 4.8a to Figure 4.8d show that our algorithm performs robustly on images not included in the training data set. That is because in our method only the data term relies on the training model, and the smoothness term is computed from the queried image itself, which reduces the effects of the training model on the parsing results. Therefore, when a query image has features not seen in the training data set, our method can use the smoothness term to correct the erroneous computation from the training model. This shows the robustness of our algorithm. Although only four test cases for such images are displayed here, the other similar test cases were performed to prove the same effectiveness of our algorithm.

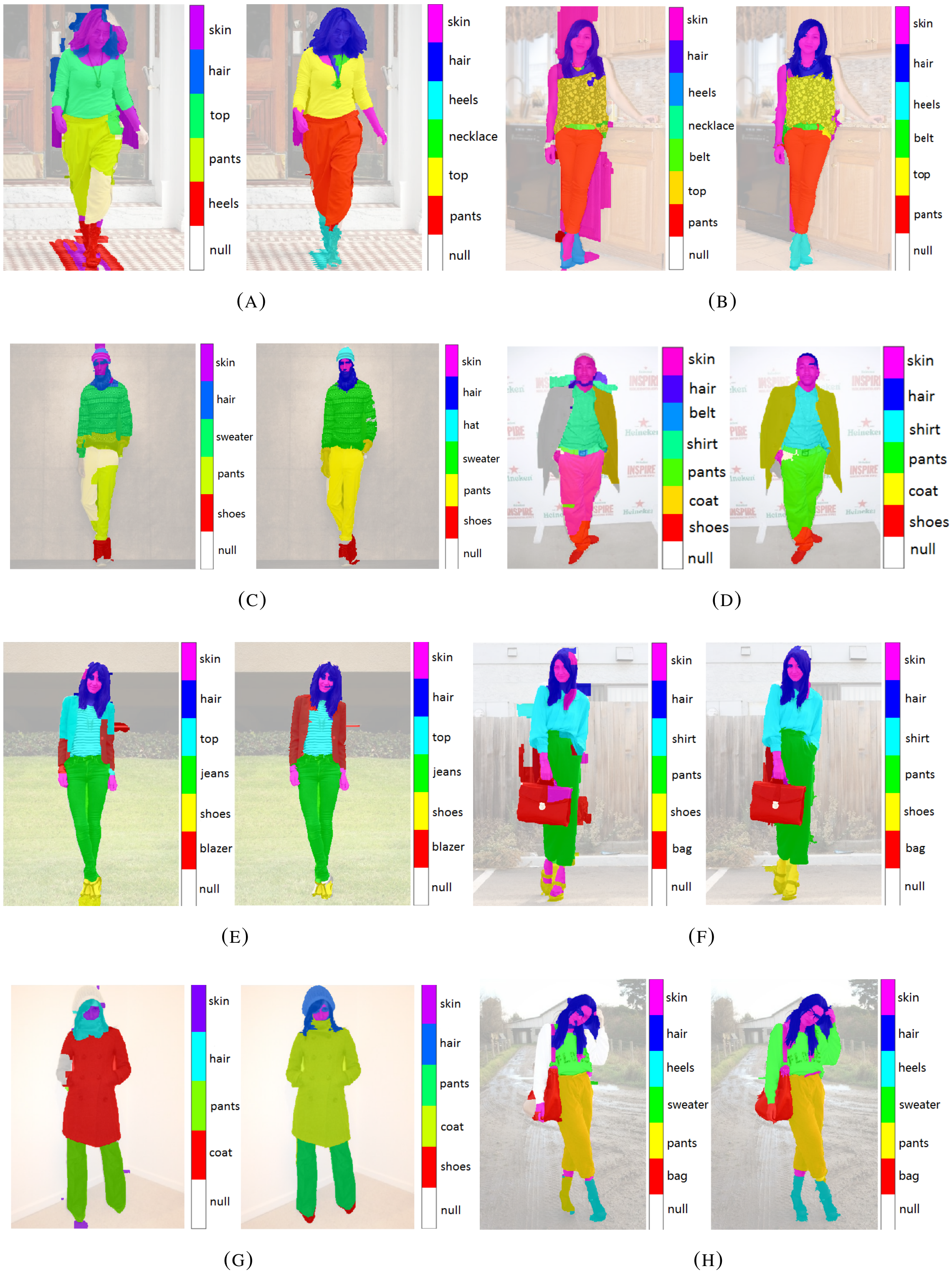
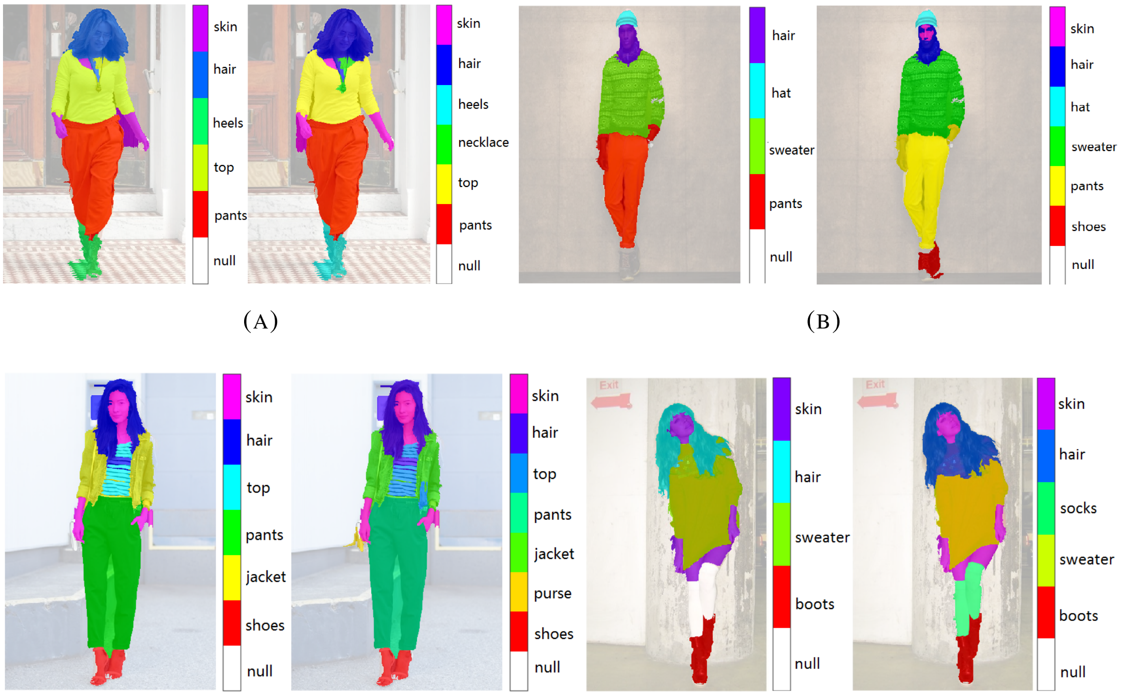


FIGURE 4.8: Comparison of image parsing results in visual quality: in each sub-figure, the left image is the parsing result by Yamaguichi *et al* [4], and the right image is our parsing result.



(A) (B)

(C) (D)

FIGURE 4.9: Comparison of parsing results in visual quality between MRF (left) and re-weighted MRF (right) model in each sub-figure.

We also compared the parsing results of re-weighted MRF versus MRF model. As shown in Figure 4.9, the necklace in (a), the shoes and skin in (b), the purse in (c), and the socks in (d) are suppressed in the MRF model, but well retained in the re-weighed MRF model.

In Figure 4.10, we illustrate progressive improvements made by using RW+BP and RW+BP+OP comparing to the existing method by Yamaguchi *et al* [4]. Images in Figure 4.10a and Figure 4.10d are illustrated in the work by Yamaguchi *et al* [4] as bad examples because of tagging spill in the background regions. As shown here, these problems can be corrected with the RW+BP+OP algorithm.

4.6 EyeDentifyIt 3.0: Automated Clothing Tagging Tool

We build an interface prototype to tag fashion images with the proposed image parsing algorithm. As shown in Figure 4.11a, for each image to be labeled, a user needs to select tags (hair, skin and null are added by default) from a fashion item list for all items

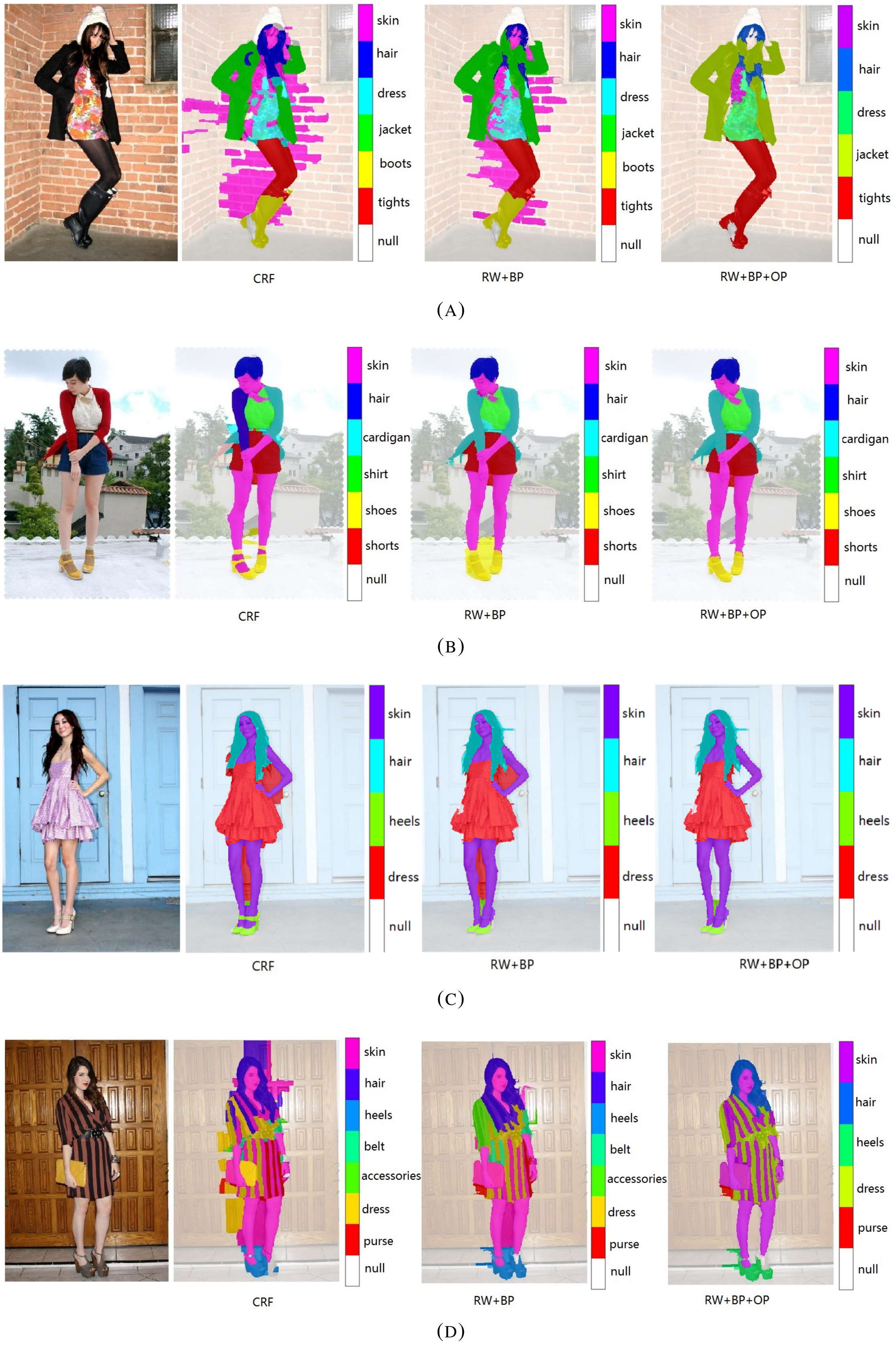
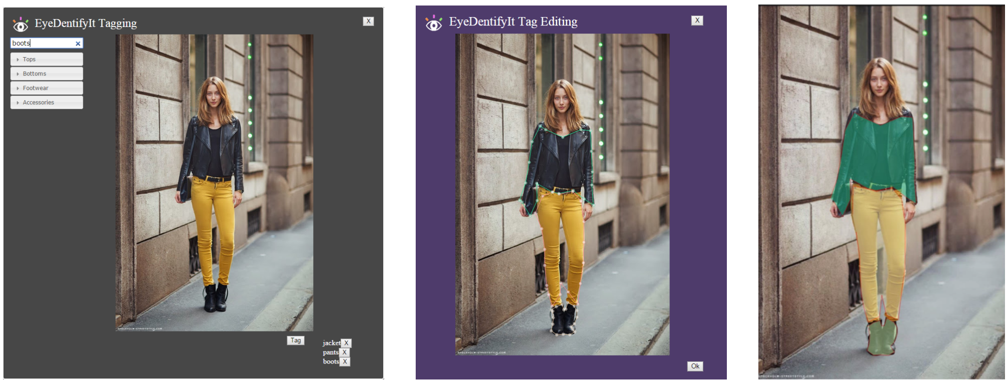


FIGURE 4.10: Comparison of parsing results in visual quality between the CRF model by Yamaguichi *et al* [4], RW+BP, and RW+BP+OP models.



(A) add tag

(B) edit tag

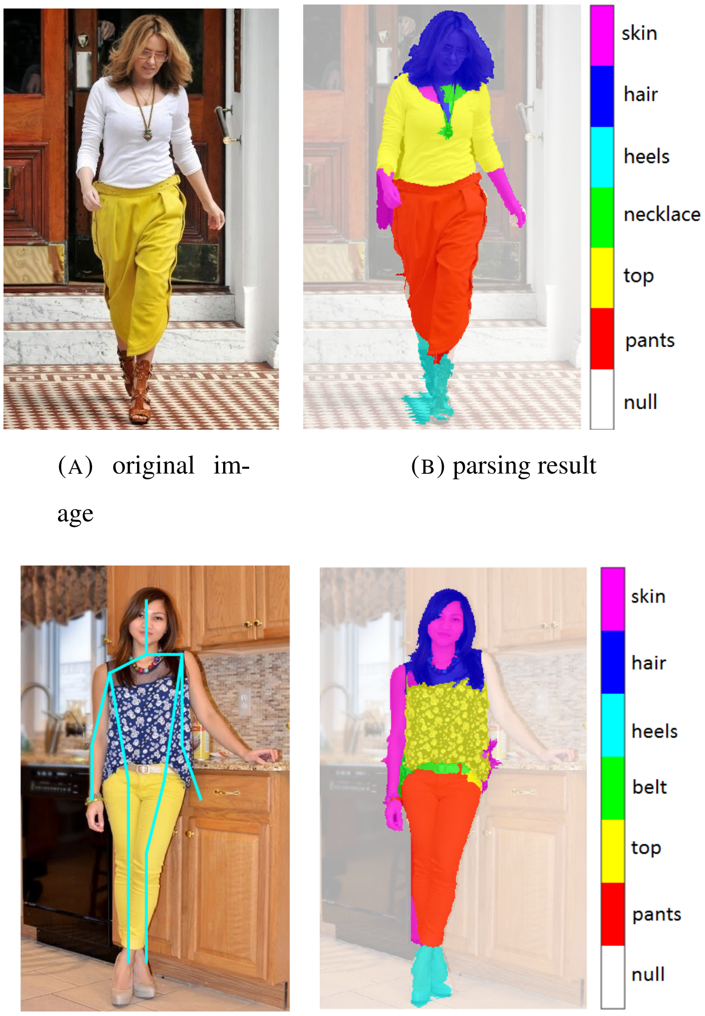
(C) tag visualization

FIGURE 4.11: New image tagging interface with automated image parsing method. Better viewed in color. (a) A user adds a list of fashion items (jacket, pants, and boots in this example) appearing in the image. (b) After computation, parsing regions for all tags are returned as different coloured polygon regions, each polygon with equally interpolated points. (c) Labeling regions can be visualized and interacted, enabled by EyeDentifyIt.

appeared in the image. The image along with the given tag list are then processed by the parsing algorithm. After computing, as shown in Figure 4.11b, the parsing region for each tag is returned as a polygon region with interpolated points, which can be edited (dragging around polygon points or adding new intermediate points) to improve the region accuracy by the user. Once satisfied with the tagging result, the user clicks "OK" to save tags along with corresponding regions in the server database for future data retrieval. Figure 4.11c shows an example that the image tagged by our tool can be visualized and interacted in an image-click-ads framework enabled by EyeDentifyIt. Comparing to state-of-the-art image labeling tools such as LabelMe [48] and Markup SVG [19], the new tagging interface saves tedious labeling process by using the image parsing method to automatically tag image regions for different tags. Although we don't pursue this further here, this interface development demonstrates the potential for integrating automatic image parsing into the image image tagging process.

4.7 Conclusion and Discussion

In this chapter, we proposed a novel method to parse fashion images into constituent garment regions that can be applied in EyeDentifyIt system. By using background prior, occlusion prior, and re-weighted MRF model we have shown that the algorithm can outperform state-of-the-art methods. The background prior initiates the border regions as the background nodes, and the occlusion prior builds NON relationship to add more edges in the graph model. Both contribute to increase the parsing accuracy. The pairwise term in our MRF model is re-defined by incorporating the data term to perform as a re-weighted pairwise term, which retains better on infrequent small region items. Our method is also robust to random query images whose features have not been seen in the training data set before. This can be explained because we employ local region contrasts to compute pairwise term which can correct erroneous data term initialization computed by the training model.



(A) original image

(B) parsing result

(C) poor pose detection

(D) parsing result

FIGURE 4.12: Illustration of limitations of the parsing method.

The limitation of our method are two folds: 1) superpixel patches of similar color tend to be over smoothed. For example, as shown in Figure 4.12b, the skin tag in the face area is merged with the hair region because the two regions have similar color. 2) Parsing results rely highly on pose detection. Because the feature vector of the superpixel patches is computed based on the relative position to the detected pose joints, poor pose detection can bias the initiation of the data term which may be too erroneous to be corrected by the pairwise term. For example, as shown in Figure 4.12d, the skin tag on the right arm area is not parsed correctly because of poor pose detection. In comparison, the CRF method also has the same limitation.

Chapter 5

Conclusion and Future Work

Advanced machine learning techniques need large scale truthful knowledge (data), such as tagged data. Currently only human can generate such accurate data used for training purposes. As mentioned in Chapter 1, existing methods of collecting large scale tagged data use either crowdsourcing and social computing. Crowdsourcing method suffers from high cost problem. Social computing is a low cost solution for large scale data collection, but it is hard to control in what direction a crowd works. In the same vein of motivating people to tag images using enjoyment incentive (e.g. ESP game), we ask the same question that weather one can motivate mass participants in social webs to work in a direction that is useful for a task assigner with low cost.

5.1 Conclusion

By examining the computing model of crowdsourcing and social computing, we explore a new model, Social Monetization Computing (SMC), that combines both. By introducing a payer to social computing, SMC model combines the motivation of crowdsourcing, which is monetary rewards, and the motivation of social computing, which is social communication. Our work suggests that mass participants can be harnessed to produce data useful for a task assigner with low cost even for free, as long as the task assigner can find a match between the wanted data and data that stimulates a social behaviour benefiting business interests.

Meanwhile, this work is the first effort on unifying three tasks of artificial intelligence under the scheme of SMC system: collecting large scale training data, developing automated technology, and tracking practical usages of automated technology.

Collecting large scale training data: The first part of this work focuses on a new data collection methodology. Overall, Chapter 2 and Chapter 3 resolved the following problems in data collection field:

- **Lower cost for a task assigner**: with additional potential payment from social webs driven by business interests, crowdsourcing workers can make overall more revenue even when a task researcher pays less task rewards. There are also free data generated by general social web users. These two ways of collecting large scale data can save significant cost for a task assigner.
- **Better incentives for a crowdsourcing worker**: crowdsourcing workers can make more revenue in SMC model comparing to crowdsourcing model.
- **Less workload**: by integrated automated technology to reduce workload horizontally and vertically, workers in SMC system take less time to generate data compared to both crowdsourcing and social computing. Therefore, SMC systems provide better user experience than crowdsourcing and social computing.

Developing automated technology: As we found free social web publishers are not very keen at using semi-automated technologies, it is necessary to integrate more automated method to further reduce workload. Chapter 4 demonstrates that by using automatic image parsing method, one can almost entirely automate the tagging process. Overall, chapter 4 resolved the following problems:

- **Automated segmentation and tagging**: integrated with image parsing technology, EyeDentifyIt can further simplifies tagging process for workers. They can simply input tagging items, then an image parsing algorithm can automatically segments an image into different regions with corresponding label assignments.
- **Efficient processing for real-time application**: traditional image parsing is computationally expensive and also has high memory cost, therefore not applicable for real-time applications like web-based tagging. Our algorithm obtains a 86.1% improvement on processing time and a 93.6% improvement on training time.

- **Higher region accuracy**: with proposed background prior, re-weighted pairwise term, and occlusion prior, our method improves image parsing accuracy by about 5.4% compared to state-of-the-art fashion parsing method.

Tracking practical usages of automated technology: By deploying automated technology in SMC systems to reduce workload for workers, interactions from all workers and users are tracked. These tracking piggy back on the SMC system is highly useful for various purposes. Although we only used tracking here for monetization purpose, a wide range of usages from such tracking data can be developed, which would be highly interesting for future work.

5.2 Future Work

Although we have built and touched all the basic elements of a SMC system, e.g. worker motivation, less cost, less workload. There are many improvements and extensions that can be done to improve the SMC system such as:

- **Piggy back on tagging**: given the current system framework, we can keep collecting on-going web-scale user interaction data with tagged data. However, such tracking data is so far not utilized to help improving automated technology. The tracking part alone can be a system piggyback on all gathered data and user interaction behaviour. Just like Google utilizes search log and user clicks of their search engine to help solve mis-spelling problem, tracking can also help solve many existing problems in current automated technology. In the future, we would like to consider how to utilize this tracking data to evaluate learning model and to automate the tagging process better.
- **Active visual learning**: there is a chicken-and-egg problem in our developed system. To build a good SMC system, we need a better automated method. Meanwhile, we need better data to learn a good model. There is significant effort [124, 125] in treating this kind of problem in human-in-the-loop framework.
- **Application of SMC system**: in this thesis, we provided a paradigm for applying SMC system guidelines by developing a prototype of image tagging system.

However, there are many existing data collection problems in different application area, e.g. natural language processing, bug reporting. Applying SMC system design guidelines to implement more applications for collecting labeled data purpose would be useful in many fields.

It is our hope that this thesis is bringing the technology of automatic image tagging one step closer to the day of becoming a reality.

Bibliography

[1] Soren Goyal and Paul Benjamin. Object recognition using deep neural networks: A survey. *arXiv preprint arXiv:1412.3684*, 2014.

[2] Luis Von Ahn. *Human Computation*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 2005. AAI3205378.

[3] Xuming He, Richard S Zemel, and Miguel Á Carreira-Perpiñán. Multiscale conditional random fields for image labeling. In *Computer vision and pattern recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE computer society conference on*, volume 2, pages II–695. IEEE, 2004.

[4] Kota Yamaguchi, M Hadi Kiapour, Luis E Ortiz, and Tamara L Berg. Parsing clothing in fashion photographs. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3570–3577. IEEE, 2012.

[5] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916, May 2011. ISSN 0162-8828.

[6] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2274–2282, November 2012. ISSN 0162-8828.

[7] `http://en.wikipedia.org/wiki/Graph_power`, 2015.

[8] ILSVRC2012. *Large Scale Visual Recognition Challenge 2012*. 2012. `http://www.image-net.org/challenges/LSVRC/2012/`.

[9] ILSVRC2014. *Large Scale Visual Recognition Challenge 2014*. 2012. `http://www.image-net.org/challenges/LSVRC/2014/`.

[10] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge, 2014.

[11] Mark Everingham, Luc Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vision*, 88(2):303–338, June 2010. ISSN 0920-5691. doi: 10.1007/s11263-009-0275-4. URL `http://dx.doi.org/10.1007/s11263-009-0275-4`.

[12] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[13] Jia Deng, Alexander C Berg, Kai Li, and Li Fei-Fei. What does classifying more than 10,000 image categories tell us? In *Computer Vision–ECCV 2010*, pages 71–84. Springer, 2010.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[15] Victor S. Sheng, Foster Provost, and Panagiotis G. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’08, pages 614–622, New York, NY, USA, 2008. ACM.

[16] Jia Deng, Olga Russakovsky, Jonathan Krause, Michael S Bernstein, Alex Berg, and Li Fei-Fei. Scalable multi-label annotation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3099–3102. ACM, 2014.

[17] Morgan Ames and Mor Naaman. Why we tag: Motivations for annotation in mobile and online media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 971–980. ACM, 2007.

[18] A. Sorokin and D. Forsyth. Utility data annotation via amazon mechanical turk. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, 2008.

[19] E. Kim, XiaoLei Huang, and Gang Tan. Markup svg: An online content-aware image abstraction and annotation tool. *Multimedia, IEEE Transactions on*, 13 (5):993–1006, Oct 2011.

[20] Stefanie Nowak and Stefan Rüger. How reliable are annotations via crowdsourcing: a study about inter-annotator agreement for multi-label image annotation. In *Proceedings of the international conference on Multimedia information retrieval*, pages 557–566. ACM, 2010.

[21] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 319–326, 2004.

[22] Alexander J. Quinn and Benjamin B. Bederson. Human computation: A survey and taxonomy of a growing field. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1403–1412, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0228-9. doi: 10.1145/1978942.1979148. URL `http://doi.acm.org/10.1145/1978942.1979148`.

[23] Panagiotis G. Ipeirotis, Foster Provost, and Jing Wang. Quality management on amazon mechanical turk. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '10, pages 64–67, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0222-7.

[24] P. Perona P. Welinder. Online crowdsourcing: rating annotators and obtaining cost-effective labels. In *CVPR*, 2010.

[25] Sirion Vittayakorn and James Hays. Quality assessment for crowdsourced object annotations. In *Proceedings of the British Machine Vision Conference*, pages 109.1–109.11. BMVA Press, 2011.

[26] Jacob Whitehill, Ting fan Wu, Jacob Bergsma, Javier R. Movellan, and Paul L. Ruvolo. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In Y. Bengio, D. Schuurmans, J.D. Lafferty, C.K.I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 2035–2043. Curran Associates, Inc., 2009.

[27] Dengyong Zhou, John C. Platt, Sumit Basu, and Yi Mao. Learning from the wisdom of crowds by minimax entropy. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.*, pages 2204–2212, 2012.

[28] Peter Welinder, Steve Branson, Serge Belongie, and Pietro Perona. The Multidimensional Wisdom of Crowds. In *NIPS*, 2010.

[29] Hao Su, Jia Deng, and Li Fei-Fei. Crowdsourcing annotations for visual object detection. In *AAAI Technical Report, 4th Human Computation Workshop*, 2012.

[30] Sudheendra Vijayanarasimhan and Kristen Grauman. Large-scale live active learning: Training object detectors with crawled data and crowds. *International Journal of Computer Vision*, 108(1-2):97–114, 2014.

[31] Ece Kamar, Severin Hacker, and Eric Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '12, pages 467–474, Richland, SC, 2012. International Foundation for Autonomous Agents and Multiagent Systems.

[32] Christopher Harris. You're Hired! An Examination of Crowdsourcing Incentive Models in Human Resource Tasks. In Matthew Lease, Vitor Carvalho, and Emine Yilmaz, editors, *Proceedings of the Workshop on Crowdsourcing for Search*

and Data Mining (CSDM) at the Fourth ACM International Conference on Web Search and Data Mining (WSDM), pages 15–18, Hong Kong, China, February 2011.

[33] Winter Mason and Duncan J. Watts. Financial incentives and the "performance of crowds". In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '09, pages 77–85, New York, NY, USA, 2009. ACM.

[34] Aniket Kittur, Bongwon Suh, Bryan A Pendleton, and Ed H Chi. He says, she says: conflict and coordination in wikipedia. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 453–462. ACM, 2007.

[35] Irwin King, Jiexing Li, and Kam Tong Chan. A brief survey of computational approaches in social computing. In *Neural Networks, 2009. IJCNN 2009. International Joint Conference on*, pages 1625–1632. IEEE, 2009.

[36] Lyndon S. Kennedy, Shih-Fu Chang, and Igor V. Kozintsev. To search or to label?: Predicting the performance of search-based automatic image classifiers. In *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, MIR '06, pages 249–258, 2006. ISBN 1-59593-495-2.

[37] Thomas W Malone, Robert Laubacher, and Chrysanthos Dellarocas. Harnessing crowds: Mapping the genome of collective intelligence. 2009.

[38] Luis Von Ahn. Games with a purpose. *Computer*, 39(6):92–94, 2006.

[39] Luis Von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. recaptcha: Human-based character recognition via web security measures. *Science*, 321(5895):1465–1468, 2008.

[40] Luis von Ahn, Ruoran Liu, and Manuel Blum. Peekaboom: A game for locating objects in images. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 55–64, 2006.

[41] J. Steggink and C. G. M. Snoek. Adding semantics to image-region annotations with the name-it-game. *Multimedia Systems*, 17(5):367–378, 2011.

[42] Chien-Ju Ho, Tao-Hsuan Chang, Jong-Chuan Lee, Jane Yung-jen Hsu, and Kuan-Ta Chen. Kisskissban: A competitive human computation game for image annotation. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '09, pages 11–14, 2009.

[43] Edith Law, Burr Settles, Aaron Snook, Harshit Surana, Luis Von Ahn, and Tom Mitchell. Human computation for attribute and attribute value acquisition. In *Proceedings of the First Workshop on Fine-Grained Visual Categorization (FGVC)*. Citeseer, 2011.

[44] Chien-Ju Ho, Tsung-Hsiang Chang, and Jane Yung-Jen Hsu. Photoslap: A multi-player online game for semantic annotation. In *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, volume 22, page 1359. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

[45] Chien-Ju Ho, Tao-Hsuan Chang, Jong-Chuan Lee, Jane Yung-jen Hsu, and Kuan-Ta Chen. Kisskissban: a competitive human computation game for image annotation. In *Proceedings of the acm sigkdd workshop on human computation*, pages 11–14. ACM, 2009.

[46] Alexa. The web information company. `http://www.alexa.com/`, 2015. [Online; accessed 2015-08-23].

[47] Dan Han, Chenlei Zhang, Xiaochao Fan, Abram Hindle, Kenny Wong, and Eleni Stroulia. Understanding android fragmentation with topic analysis of vendor-specific bugs. In *Reverse Engineering (WCRE), 2012 19th Working Conference on*, pages 83–92. IEEE, 2012.

[48] Bryan C. Russell, Antonio Torralba, Kevin P. Murphy, and William T. Freeman. Labelme: A database and web-based tool for image annotation. *Int. J. Comput. Vision*, 77(1-3):157–173, May 2008. ISSN 0920-5691.

[49] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 309–316, Sept 2009.

[50] Yasuhide Mori, Hironobu Takahashi, and Ryuichi Oka. Image-to-word transformation based on dividing and vector quantizing images with words. In *First International Workshop on Multimedia Intelligent Storage and Retrieval Management*. Citeseer, 1999.

[51] Pinar Duygulu, Kobus Barnard, Joao FG de Freitas, and David A Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Computer Vision—ECCV 2002*, pages 97–112. Springer, 2002.

[52] Kobus Barnard, Pinar Duygulu, David Forsyth, Nando De Freitas, David M. Blei, Jaz K, Thomas Hofmann, Tomaso Poggio, and John Shawe-taylor. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.

[53] Florent Monay and Daniel Gatica-Perez. Plsa-based image auto-annotation: Constraining the latent space. In *Proceedings of the 12th Annual ACM International Conference on Multimedia*, MULTIMEDIA ’04, pages 348–351, New York, NY, USA, 2004. ACM. ISBN 1-58113-893-8. doi: 10.1145/1027527.1027608. URL `http://doi.acm.org/10.1145/1027527.1027608`.

[54] Oksana Yakhnenko and Vasant Honavar. Annotating images and image objects using a hierarchical dirichlet process model. In *Proceedings of the 9th International Workshop on Multimedia Data Mining: Held in Conjunction with the ACM SIGKDD 2008*, MDM ’08, pages 1–7. ACM, 2008. ISBN 978-1-60558-261-0. doi: 10.1145/1509212.1509213. URL `http://doi.acm.org/10.1145/1509212.1509213`.

[55] P. Duygulu, Kobus Barnard, J. F. G. de Freitas, and David A. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proceedings of the 7th European Conference on Computer Vision-Part IV*, ECCV ’02, pages 97–112, London, UK, UK, 2002. Springer-Verlag.

ISBN 3-540-43748-7. URL `http://dl.acm.org/citation.cfm?id=645318.649254`.

[56] G. Carneiro, AB. Chan, P.J. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(3):394–410, March 2007.

[57] S.L. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–1002–II–1009 Vol.2, June 2004.

[58] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, pages 119–126, New York, NY, USA, 2003. ACM. ISBN 1-58113-646-3. doi: 10.1145/860435.860459. URL `http://doi.acm.org/10.1145/860435.860459`.

[59] Victor Lavrenko, R. Manmatha, and Jiwoon Jeon. A model for learning the semantics of pictures. In S. Thrun, L.K. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pages 553–560. MIT Press, 2004.

[60] D. Grangier and S. Bengio. A discriminative kernel-based approach to rank images from text queries. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(8):1371–1384, Aug 2008.

[61] Tomer Hertz, Aharon Bar-hillel, and Daphna Weinshall. Learning distance functions for image retrieval. In *In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 570–577, 2004.

[62] Bangalore S Manjunath and Wei-Ying Ma. Texture features for browsing and retrieval of image data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(8):837–842, 1996.

[63] Wei-Ying Ma and Bangalore S Manjunath. Netra: A toolbox for navigating large image databases. *Multimedia systems*, 7(3):184–198, 1999.

[64] John R Smith and Shih-Fu Chang. Transform features for texture classification and discrimination in large image databases. In *Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference*, volume 3, pages 407–411. IEEE, 1994.

[65] Jamie Shotton, Matthew Johnson, and Roberto Cipolla. Semantic texton forests for image categorization and segmentation. In *Computer vision and pattern recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[66] Andrzej Materka, Michal Strzelecki, et al. Texture analysis methods–a review. *Technical university of lodz, institute of electronics, COST B11 report, Brussels*, pages 9–11, 1998.

[67] Olivier Chapelle, Patrick Haffner, and Vladimir N Vapnik. Support vector machines for histogram-based image classification. *Neural Networks, IEEE Transactions on*, 10(5):1055–1064, 1999.

[68] Claudio Cusano, Gianluigi Ciocca, and Raimondo Schettini. Image annotation using svm. In *Electronic Imaging 2004*, pages 330–338. International Society for Optics and Photonics, 2003.

[69] Fabio Del Frate, Fabio Pacifici, Giovanni Schiavon, and Chiara Solimini. Use of neural networks for automatic classification from high-resolution images. *Geoscience and Remote Sensing, IEEE Transactions on*, 45(4):800–809, 2007.

[70] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[71] J Ross Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.

[72] Hao Zhang, Alexander C. Berg, Michael Maire, and Jitendra Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, CVPR '06, pages 2126–2136. IEEE

Computer Society, 2006. ISBN 0-7695-2597-0. doi: 10.1109/CVPR.2006.301. URL `http://dx.doi.org/10.1109/CVPR.2006.301`.

[73] Ameesh Makadia, Vladimir Pavlovic, and Sanjiv Kumar. A new baseline for image annotation. In *Proceedings of the 10th European Conference on Computer Vision: Part III*, ECCV '08, pages 316–329, 2008.

[74] Shaoting Zhang, Junzhou Huang, Yuchi Huang, Yang Yu, Hongsheng Li, and D.N. Metaxas. Automatic image annotation using group sparsity. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3312–3319, June 2010.

[75] Xin-Jing Wang, Lei Zhang, Xirong Li, and Wei-Ying Ma. Annotating images by mining image search results. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(11): 1919–1932, November 2008.

[76] Lei Wu, Steven C.H. Hoi, Rong Jin, Jianke Zhu, and Nenghai Yu. Distance metric learning from uncertain side information with application to automated photo tagging. In *Proceedings of the 17th ACM International Conference on Multimedia*, MM '09, pages 135–144, 2009.

[77] Theodora Tsikrika, Christos Diou, Arjen P. de Vries, and Anastasios Delopoulos. Image annotation using clickthrough data. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, CIVR '09, pages 14:1–14:8, New York, NY, USA, 2009. ACM.

[78] Christian Fluhr, Pierre-Alain Moëllic, and Patrick Hede. Usage-oriented multimedia information retrieval technological evaluation. In *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, MIR '06, pages 301–306. ACM, 2006. ISBN 1-59593-495-2.

[79] Li Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(4):594–611, April 2006.

[80] Scott Krig. Survey of ground truth datasets. In *Computer Vision Metrics*, pages 401–410. Springer, 2014.

[81] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007.

[82] Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part I*, ECCV'06, pages 1–15, Berlin, Heidelberg, 2006. Springer-Verlag. ISBN 3-540-33832-2, 978-3-540-33832-1. doi: 10.1007/11744023_1. URL `http://dx.doi.org/10.1007/11744023_1`.

[83] Antonio Torralba. Contextual priming for object detection. *Int. J. Comput. Vision*, 53(2):169–191, July 2003. ISSN 0920-5691. doi: 10.1023/A:1023052124951. URL `http://dx.doi.org/10.1023/A:1023052124951`.

[84] Derek Hoiem, Alexei A. Efros, and Martial Hebert. Putting objects in perspective. *Int. J. Comput. Vision*, 80(1):3–15, October 2008. ISSN 0920-5691. doi: 10.1007/s11263-008-0137-5. URL `http://dx.doi.org/10.1007/s11263-008-0137-5`.

[85] Y.Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary amp; region segmentation of objects in n-d images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 105–112 vol.1, 2001.

[86] Yin Li, Jian Sun, Chi-Keung Tang, and Heung-Yeung Shum. Lazy snapping. *ACM Trans. Graph.*, 23(3):303–308, August 2004. ISSN 0730-0301.

[87] Stuart Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-6(6):721–741, Nov 1984.

[88] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(11):1222–1239, November 2001. ISSN 0162-8828.

[89] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[90] `http://www.image-net.org/challenges/LSVRC/2010/`, 2010.

[91] Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press, 1998.

[92] D.G. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157 vol.2, 1999.

[93] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[94] Christoph Zauner. Implementation and benchmarking of perceptual image hash functions. *Master's thesis*, 2010.

[95] K Ramamohan Rao and Ping Yip. *Discrete cosine transform: algorithms, advantages, applications*. Academic press, 2014.

[96] Ronny Lempel and Aya Soffer. Picashow: Pictorial authority search by hyperlinks on the web. In *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pages 438–448, New York, NY, USA, 2001. ACM. ISBN 1-58113-348-0.

[97] Lee R Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.

[98] Si Liu, Jiashi Feng, C. Domokos, Hui Xu, Junshi Huang, Zhenzhen Hu, and Shuicheng Yan. Fashion parsing with weak color-category labels. *Multimedia, IEEE Transactions on*, 16(1):253–265, Jan 2014.

[99] Zhuowen Tu, Xiangrong Chen, Alan L Yuille, and Song-Chun Zhu. Image parsing: Unifying segmentation, detection, and recognition. *International Journal of computer vision*, 63(2):113–140, 2005.

[100] Thomas Leung and Jitendra Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International journal of computer vision*, 43(1):29–44, 2001.

[101] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip Torr. Conditional random fields as recurrent neural networks. *arXiv preprint arXiv:1502.03240*, 2015.

[102] John Winn and Nebojsa Jojic. Locus: Learning object classes with unsupervised segmentation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 756–763. IEEE, 2005.

[103] Joseph Tighe and Svetlana Lazebnik. Superparsing: scalable nonparametric image parsing with superpixels. In *Computer Vision–ECCV 2010*, pages 352–365. Springer, 2010.

[104] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):721–741, 1984.

[105] Stan Z Li. *Markov random field modeling in image analysis*. Springer Science & Business Media, 2009.

[106] M. Weber, M. Bauml, and R. Stiefelhagen. Part-based clothing segmentation for person retrieval. In *Proceedings of the 2011 8th IEEE International Conference on Advanced Video and Signal Based Surveillance*, AVSS '11, pages 361–366, Washington, DC, USA, 2011. IEEE Computer Society.

[107] Nils Hasler, Carsten Stoll, Bodo Rosenhahn, Thorsten Thormählen, and Hans-Peter Seidel. Technical section: Estimating body shape of dressed humans. *Comput. Graph.*, 33(3):211–216, June 2009. ISSN 0097-8493.

[108] Huizhong Chen, Andrew Gallagher, and Bernd Girod. Describing clothing by semantic attributes. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part III*, ECCV'12, pages 609–623, Berlin, Heidelberg, 2012. Springer-Verlag.

[109] Lukas Bossard, Matthias Dantone, Christian Leistner, Christian Wengert, Till Quack, and Luc Van Gool. Apparel classification with style. In *Proceedings of the 11th Asian Conference on Computer Vision - Volume Part IV*, ACCV'12, pages 321–335, Berlin, Heidelberg, 2013. Springer-Verlag.

[110] L. Bourdev, S. Maji, and J. Malik. Describing people: A poselet-based approach to attribute classification. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, Nov 2011.

[111] Si Liu, Jiashi Feng, Zheng Song, Tianzhu Zhang, Hanqing Lu, Changsheng Xu, and Shuicheng Yan. Hi, magic closet, tell me what to wear! In *Proceedings of the 20th ACM International Conference on Multimedia*, MM '12, pages 619–628, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1089-5. doi: 10.1145/2393347.2393433. URL `http://doi.acm.org/10.1145/2393347.2393433`.

[112] Zheng Song, Meng Wang, Xian-Sheng Hua, and Shuicheng Yan. Predicting occupation via human clothing and contexts. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1084–1091, Nov 2011.

[113] Ana C. Murillo, Iljung S. Kwak, Lubomir D. Bourdev, David J. Kriegman, and Serge Belongie. Urban tribes: Analyzing group photos from a social perspective. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, June 16-21, 2012*, pages 28–35, 2012.

[114] Basela Hasan and David Hogg. Segmentation using deformable spatial priors with application to clothing. In *Proceedings of the British Machine Vision Conference*, pages 83.1–83.11. BMVA Press, 2010.

[115] Jian Dong, Qiang Chen, Wei Xia, Zhongyang Huang, and Shuicheng Yan. A deformable mixture parsing model with parselets. In *Proceedings of the 2013 IEEE International Conference on Computer Vision*, ICCV '13, pages 3408–3415, Washington, DC, USA, 2013. IEEE Computer Society. ISBN 978-1-4799-2840-8.

[116] K. Yamaguchi, M.H. Kiapour, and T.L. Berg. Paper doll parsing: Retrieving similar styles to parse clothing items. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3519–3526, Dec 2013.

[117] Hanqing Lu, Changsheng Xu, Guangcan Liu, Zheng Song, Si Liu, and Shuicheng Yan. Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 0:3330–3337, 2012. ISSN 1063-6919.

[118] Yannis Kalantidis, Lyndon Kennedy, and Li-Jia Li. Getting the look: Clothing recognition and segmentation for automatic product suggestions in everyday photos. In *Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval*, ICMR '13, pages 105–112, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2033-7.

[119] Yi Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pages 1385–1392, Washington, DC, USA, 2011. IEEE Computer Society. ISBN 978-1-4577-0394-2.

[120] Yichen Wei, Fang Wen, Wangjiang Zhu, and Jian Sun. Geodesic saliency using background priors. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part III*, ECCV'12, pages 29–42, Berlin, Heidelberg, 2012. Springer-Verlag. ISBN 978-3-642-33711-6.

[121] János Schanda. *Colorimetry: Understanding the CIE system.* John Wiley & Sons, 2007.

[122] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9: 1871–1874, June 2008. ISSN 1532-4435.

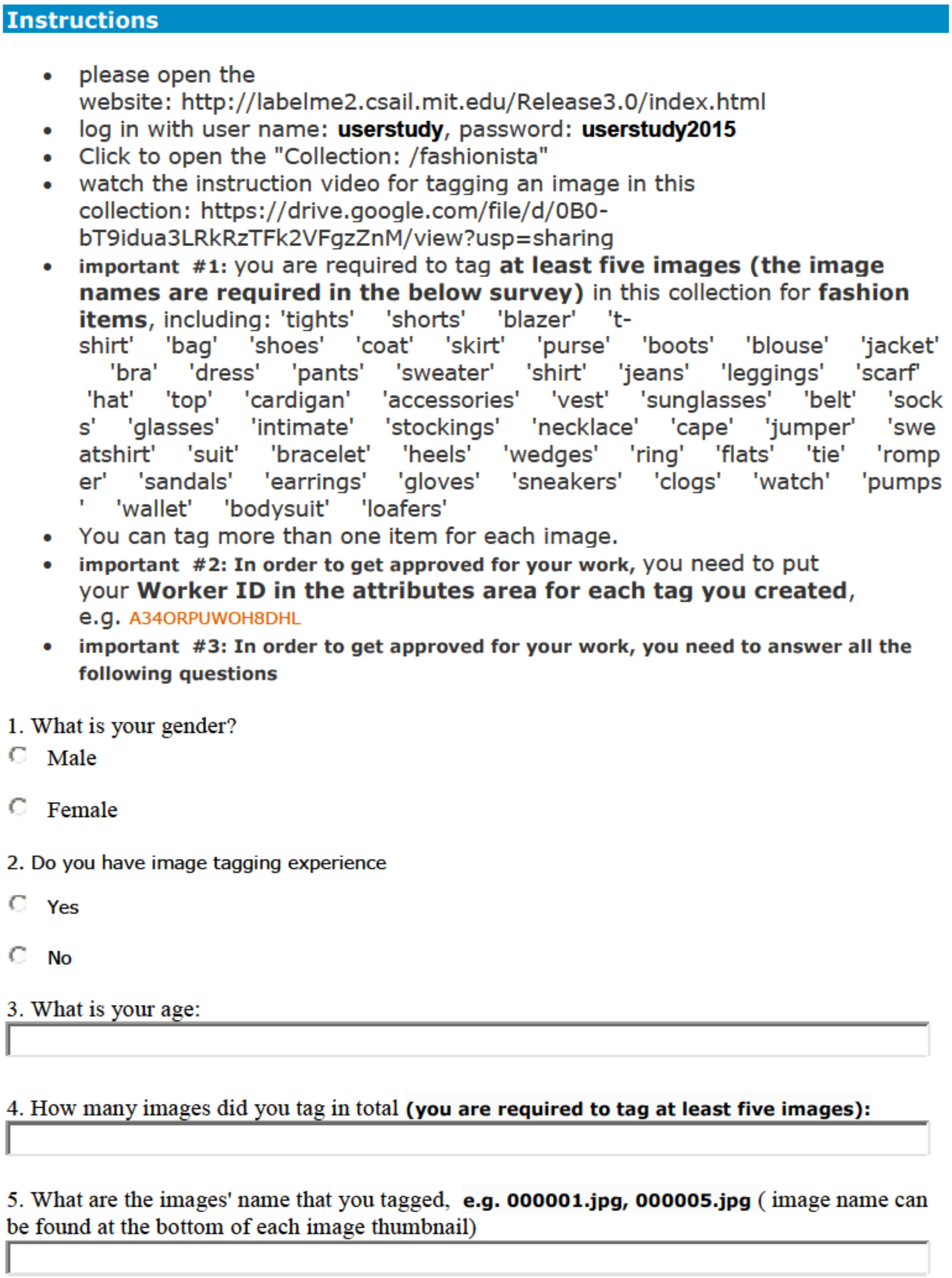
[123] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(9):1124–1137, September 2004.

[124] *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, 2014. IEEE.

[125] Suyog Dutt Jain and Kristen Grauman. Predicting sufficient annotation strength for interactive foreground segmentation. *The IEEE International Conference on Computer Vision (ICCV)*, December 2013.

Appendix A: Usability Study on Amazon Mechanical Turk

.1 Questionnaire



Instructions

- please open the website: http://labelme2.csail.mit.edu/Release3.0/index.html
- log in with user name: **userstudy**, password: **userstudy2015**
- Click to open the "Collection: /fashionista"
- watch the instruction video for tagging an image in this collection: https://drive.google.com/file/d/0B0-bT9idua3LRkRzTFk2VFgzZnM/view?usp=sharing
- **important #1:** you are required to tag **at least five images (the image names are required in the below survey)** in this collection for **fashion items**, including: 'tights' 'shorts' 'blazer' 't-shirt' 'bag' 'shoes' 'coat' 'skirt' 'purse' 'boots' 'blouse' 'jacket' 'bra' 'dress' 'pants' 'sweater' 'shirt' 'jeans' 'leggings' 'scarf' 'hat' 'top' 'cardigan' 'accessories' 'vest' 'sunglasses' 'belt' 'socks' 'glasses' 'intimate' 'stockings' 'necklace' 'cape' 'jumper' 'sweatshirt' 'suit' 'bracelet' 'heels' 'wedges' 'ring' 'flats' 'tie' 'romper' 'sandals' 'earrings' 'gloves' 'sneakers' 'clogs' 'watch' 'pumps' 'wallet' 'bodysuit' 'loafers'
- You can tag more than one item for each image.
- **important #2: In order to get approved for your work,** you need to put your **Worker ID in the attributes area for each tag you created**, e.g. A34ORPUWOH8DHL
- **important #3: In order to get approved for your work, you need to answer all the following questions**

1. What is your gender?

○ Male

○ Female

2. Do you have image tagging experience

○ Yes

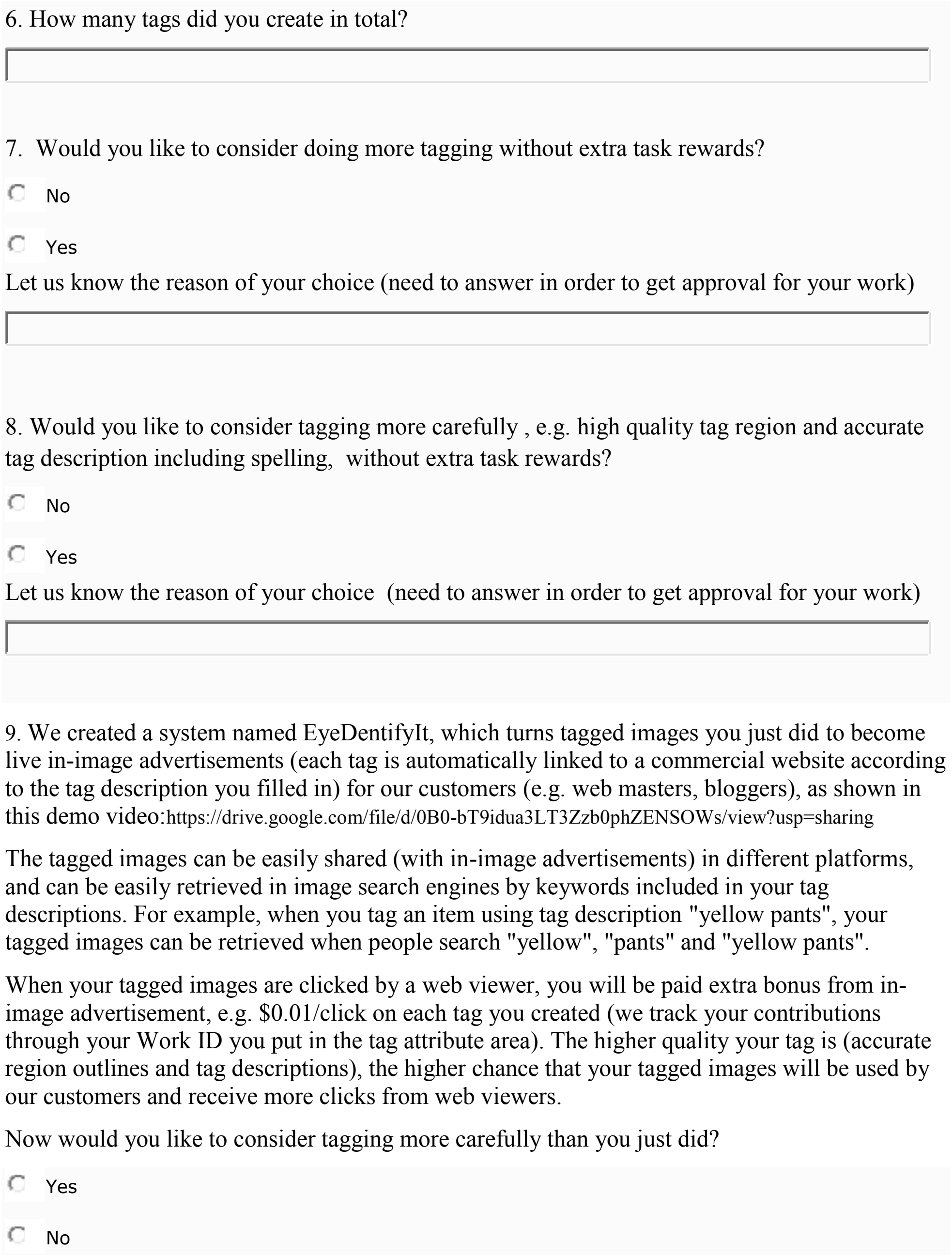
○ No

3. What is your age:

4. How many images did you tag in total **(you are required to tag at least five images):**

5. What are the images' name that you tagged, **e.g. 000001.jpg, 000005.jpg** (image name can be found at the bottom of each image thumbnail)

.2 Questionnaire



6. How many tags did you create in total?

7. Would you like to consider doing more tagging without extra task rewards?

○ No

○ Yes

Let us know the reason of your choice (need to answer in order to get approval for your work)

8. Would you like to consider tagging more carefully , e.g. high quality tag region and accurate tag description including spelling, without extra task rewards?

○ No

○ Yes

Let us know the reason of your choice (need to answer in order to get approval for your work)

9. We created a system named EyeDentifyIt, which turns tagged images you just did to become live in-image advertisements (each tag is automatically linked to a commercial website according to the tag description you filled in) for our customers (e.g. web masters, bloggers), as shown in this demo video:https://drive.google.com/file/d/0B0-bT9idua3LT3Zzb0phZENSOWs/view?usp=sharing

The tagged images can be easily shared (with in-image advertisements) in different platforms, and can be easily retrieved in image search engines by keywords included in your tag descriptions. For example, when you tag an item using tag description "yellow pants", your tagged images can be retrieved when people search "yellow", "pants" and "yellow pants".

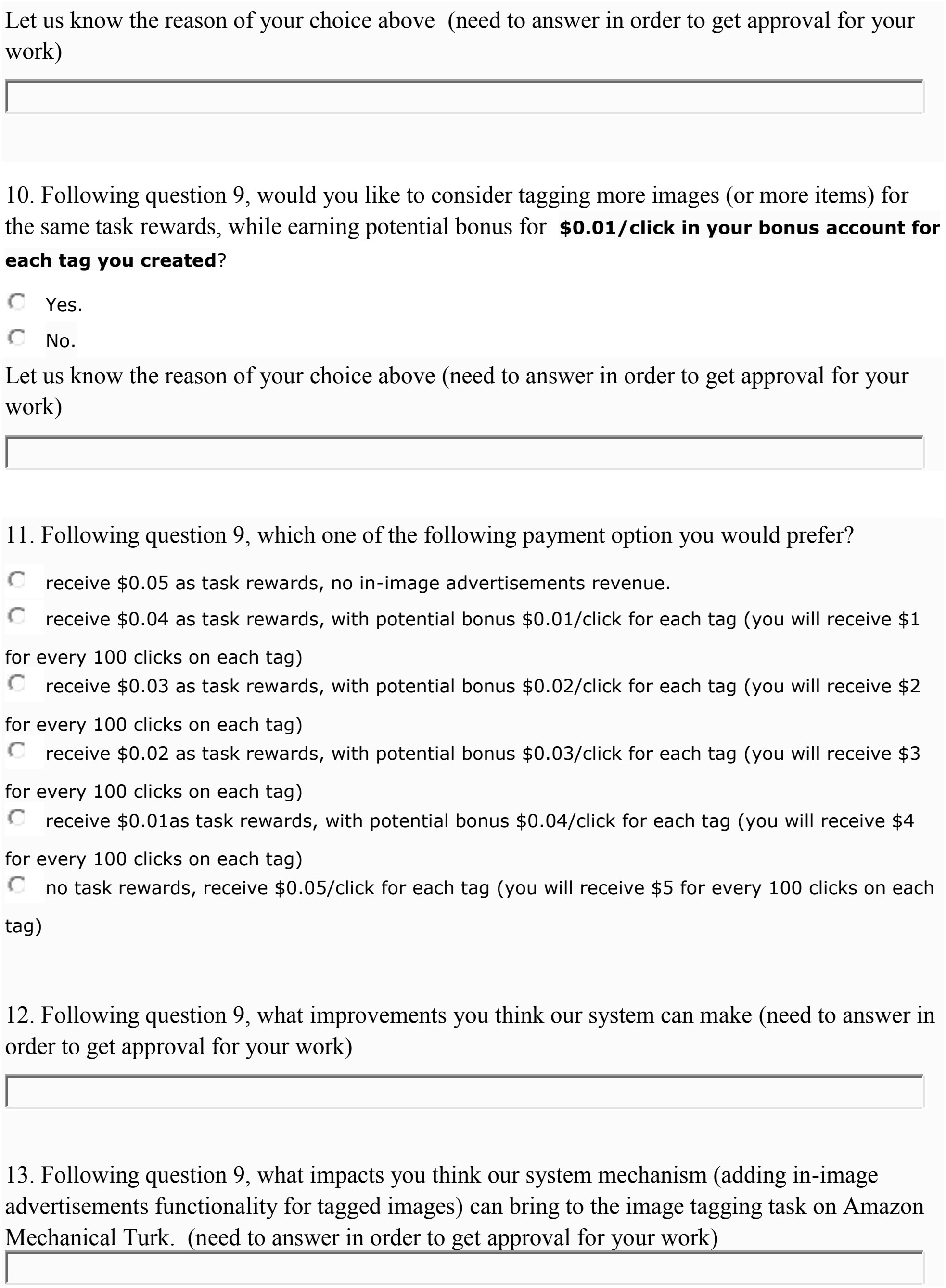
When your tagged images are clicked by a web viewer, you will be paid extra bonus from in-image advertisement, e.g. $0.01/click on each tag you created (we track your contributions through your Work ID you put in the tag attribute area). The higher quality your tag is (accurate region outlines and tag descriptions), the higher chance that your tagged images will be used by our customers and receive more clicks from web viewers.

Now would you like to consider tagging more carefully than you just did?

○ Yes

○ No

.3 Questionnaire



Let us know the reason of your choice above (need to answer in order to get approval for your work)

10. Following question 9, would you like to consider tagging more images (or more items) for the same task rewards, while earning potential bonus for **$0.01/click in your bonus account for each tag you created**?

○ Yes.

○ No.

Let us know the reason of your choice above (need to answer in order to get approval for your work)

11. Following question 9, which one of the following payment option you would prefer?

○ receive $0.05 as task rewards, no in-image advertisements revenue.

○ receive $0.04 as task rewards, with potential bonus $0.01/click for each tag (you will receive $1 for every 100 clicks on each tag)

○ receive $0.03 as task rewards, with potential bonus $0.02/click for each tag (you will receive $2 for every 100 clicks on each tag)

○ receive $0.02 as task rewards, with potential bonus $0.03/click for each tag (you will receive $3 for every 100 clicks on each tag)

○ receive $0.01as task rewards, with potential bonus $0.04/click for each tag (you will receive $4 for every 100 clicks on each tag)

○ no task rewards, receive $0.05/click for each tag (you will receive $5 for every 100 clicks on each tag)

12. Following question 9, what improvements you think our system can make (need to answer in order to get approval for your work)

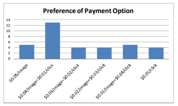
13. Following question 9, what impacts you think our system mechanism (adding in-image advertisements functionality for tagged images) can bring to the image tagging task on Amazon Mechanical Turk. (need to answer in order to get approval for your work)

.4 Usability Study Data

HITId	CreationTime	MaxAssignments	AssignmentId	WorkerId	AssignmentStatus	AcceptTime	SubmitTime	AutoApprovalTime	WorkTimeInSeconds	Answer.#1. Gender	Answer.#10. after video tag more?	Answer.#10. after video tag more? reason	Answer.#11. payment plan	Answer.#12. improvements	Answer.#13. comments about image-click-ads impact	Answer.#2. tag experience	Answer.#3. age	Answer.#4. image #	Answer.#5. image names	Answer.#6. tag #	Answer.#7. tag more without extra pay?	Answer.#7. tag more without extra pay? reason	Answer.#8. tag more carefully?	Answer.#8. tag more carefully? reason	Answer.#9. after video tag more carefully?	Answer.#9. after video tag more carefully? reason
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3H7Z272LX76YKODOSTGW18BADW7LPU	AOSB3RSWQQENC	Approved	Tue Jan 06 03:17:05 GMT 2015	Tue Jan 06 03:46:46 GMT 2015	Thu Jan 08 19:46:46 PST 2015	1781	Female	Yes	very interesting work. It can bring more income for workers	$0.05/image	Polyfon tool needs to be finer. Something like lasso tool of photoshop would be nice.	Can be helpful to workers.	No	59	5	000451.jpg (purse) 000013.jpg (skirt) 000303.jpg (socks) 000547.jpg (shoes) 000525.jpg (hat)	5	No		No		Yes	it may bring potential revenue
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3DY46V3X3PHGPPSKGRRBMDAW06G55E	A2V0Y764TM0O2Q	Approved	Tue Jan 06 05:06:45 GMT 2015	Tue Jan 06 05:17:15 GMT 2015	Thu Jan 08 21:17:15 PST 2015	630	Female	Yes		0.03/image+0.02/per click	Better design of the labelme site, its harsh on the eyes	None	Yes	24	5	005539.jpg, 003758.jpg, 004307.jpg, 003750.jpg, 000525.jpg	5	No	Need more pay, I enjoyed doing it though, reminds me of Polyvore I used to be huge in that community or LookBook.nu	No		Yes	
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3U8YCDAGXPF6N4PH3D923T9WFUY0Q7	A45YWWUL5LSLI	Approved	Tue Jan 06 06:17:54 GMT 2015	Tue Jan 06 07:07:14 GMT 2015	Thu Jan 08 23:07:14 PST 2015	2960	Female	Yes	I like the work and if the payment is good I'll do a lot more.	0.02/image+0.03/per click	I don't have any.	I want more hits.	No	26	8	OO3206.jpg, OO3212.jpg, OO3240.jpg, OO3255.jpg, OO3425.jpg, OO3461.jpg, OO3491.jpg, OO3545.jpg,	8	No	Yes, because the task is interesting, but only for payment.	No	I prefer to do everything as best as I can, and I already did my best.	No	I already did my best.
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3LYA37P8IQMLWOUWOOMUROT67S6BKF	ANS32KK8E96UK	Approved	Tue Jan 06 06:52:38 GMT 2015	Tue Jan 06 07:14:05 GMT 2015	Thu Jan 08 23:14:05 PST 2015	1287	Male	Yes		0.04/image+0.01/per click	good	good	No	30	6	000528.jpg,000129.jpg,000553.jpg,000708.jpg,000739.jpg,000934.jpg	6	No	no compensation for extra work	No	takes is too tedious	Yes	
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	UWOOMUROT67S6BKF	ANS32KK8E96UK	Approved	Tue Jan 06 06:52:38 GMT 2015	07:14:05 GMT 2015	Thu Jan 08 23:14:05 PST 2015	1287	Male	Yes		0.04/image+0.01/per click	good	good	No	30	6	000528.jpg,000129.jpg,000553.jpg,000708.jpg,000739.jpg,000934.jpg	6	No		No		Yes	
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3DIP6YHAPCRZ8Q4FPQRH2RQ0FQX8E3	A2V2RHD14OE4XY	Approved	Tue Jan 06 07:56:45 GMT 2015	Tue Jan 06 08:16:35 GMT 2015	Fri Jan 09 00:16:35 PST 2015	1190	Male	Yes	This is something that I would certainly consider depending on how often these images are clicked. I also feel that my tagging will get more precise with practice	0.02/image+0.03/per click	I think that it is laid out very well, I would consider doing more tagging given the information that I was provided with.	I liked this task, I was unaware of image-click-ads before this survey but now I am interested in learning more.	Yes	25	5	002384.jpg, 002417.jpg, 003088.jpg, 003158.jpg, 000430.jpg	20	No	I would consider doing this if there was a form of compensation	No	I believe that you should do the work to the best of your ability, especially if you expect compensation. Take pride in your work.	No	I feel like my tags are accurate
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3K2755HG5S23VZ0219M361P12I9FDY	A1IWHXML03XWSQ	Approved	Tue Jan 06 07:55:02 GMT 2015	Tue Jan 06 08:41:34 GMT 2015	Fri Jan 09 00:41:34 PST 2015	2792	Male	Yes	because I need not work for my extra compensations	0.02/image+0.03/per click		It will be of huge impact if it starts to produce money like this.	No	23	5	001885.jpg, 002071.jpg, 002603.jpg, 002563.jpg, 006004.jpg	17	No		No	takes too much time	No	I've already done tagging carefully enough for that purpose
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3OXV7EAXLEP9UE5U7RDNZMQCI6O36S	A2VN59CZEQ7P0R	Approved	Tue Jan 06 11:10:35 GMT 2015	Tue Jan 06 11:39:43 GMT 2015	Fri Jan 09 03:39:43 PST 2015	1748	Male	Yes		0.05/per click	The "per-click" payment system seems fair and a good incentive for quality work. I am concerned about people taking advantage and abusing it, however.	There are other image tagging HITS out there on MTurk, but they have high expectations for very low pay. If you offer a fair wage for quality work, or use a qualification system to get quality workers, it would be to your benefit.	No	35	6	000136.jpg, 000247.jpg, 000251.jpg, 000509.jpg, 000525.jpg, 000533.jpg	6	No	yes, it was interesting, but only for compensation	No		No	I did fairly accurately the first time. I will be more accurate if given another opportunity.
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3A9AA95ATWLKIZA3060PKJFDR8F5P2	AWL97LXLETUVR	Approved	Tue Jan 06 22:23:11 GMT 2015	Tue Jan 06 22:34:42 GMT 2015	Fri Jan 09 14:34:42 PST 2015	691	Female	Yes		0.04/image+0.01/per click			No	23	5	002129, 002001, 000294, 001541, 002124	6	No		No	tag perfect polygon is not an easy task	Yes	
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3E4GGUZ1T8QR8BU05VTJQVQ2DR1K24	A1STXCESBVVV1J	Approved	Thu Jan 08 01:59:17 GMT 2015	Thu Jan 08 02:27:42 GMT 2015	Sat Jan 10 18:27:42 PST 2015	1705	Male	Yes		0.01/image+0.04/per click	Task is fine.		No	34	5	001923.jpg, 001861.jpg, 001765.jpg, 002064.jpg, 002440.jpg	7	No		No		Yes	
3T2EL38U0LJBXAYYFNT7NN1ODIOQX9	Mon Jan 05 22:40:58 GMT 2015	31	3J2UYBXQQLBDDM6XO4V4XOS5FZ9069	A1BHZGNIXAGM17	Approved	Fri Jan 09 22:16:36 GMT 2015	Fri Jan 09 22:39:20 GMT 2015	Mon Jan 12 14:39:20 PST 2015	1364	Female	Yes	I would be interested in earning more for tagging multiple pictures.	$0.05/image	make actual tagging instructions more specific (ex. do you want "sweater" or "top" or "shirt" as a tag?)	I think payment for clicks is not a good incentive for me, because I know nothing about the volume on the website or the interests of people who visit it. The task was interesting.	Yes	22	5	001914.jpg, 002071.jpg, 007094.jpg, 007128.jpg, 003558.jpg	11	No		No	I would rather work on other HITS that do pay for more work.	No	The general area of the tagged item will be enough to get the consumer to the right area, so very specific, high-quality tagged regions are not necessary.
3FCO4VKOZ3CSLS22OJU9Z7ZM1LX7E2	Fri Aug 21 14:45:05 PDT 2015	10	3Q5C1WP23M0H17R1F4PZJNXG39D15D	A1T29BTOVBSEA6	Approved	Fri Aug 21 14:59:42 PDT 2015	Fri Aug 21 15:14:03 PDT 2015	Mon Aug 24 15:14:03 PDT 2015	861	Male	Yes	Yes, the pay would be quite well worth it.	0.02/image+0.03/per click	Interesting idea. Could tagging be broken into separate HITs, so that users can turn in more than one task?	I'd be curious to know whether it could really work that way. Tasks for ongoing pay as users continue to interact sounds very promising. I could see myself doing quite a lot of image tagging.	No	32	5	000556.jpg, 000621.jpg, 000708.jpg, 000430.jpg, 000451.jpg	5	No	Not without extra rewards.	No	I tagged regions carefully and accurately, but I wouldn't increase effort without rewards.	Yes	I'm working for pay--now that I see the questions above were tricky, andf
3FCO4VKOZ3CSLS22OJU9Z7ZM1LX7E2	Fri Aug 21 14:45:05 PDT 2015	10	3UOUJI6MTDD6CN29NYIEM8TX75JXUH	A3EJI3R6ZE9Y4X	Approved	Fri Aug 21 15:00:10 PDT 2015	Fri Aug 21 15:12:09 PDT 2015	Mon Aug 24 15:12:09 PDT 2015	719	Male	Yes	I would tag lots of images in that case.	0.05/per click	I think the system works fine as-is.	I don't know.	Yes	32	5	000001.jpg, 000860.jpg, 000750.jpg, 000739.jpg, 000715.jpg	6	No	I would rather work on other HITS that do pay for more work.	No	I would rather work on other HITS that do pay for more work.	No	I tagged carefully.
3FCO4VKOZ3CSLS22OJU9Z7ZM1LX7E2	Fri Aug 21 14:45:05 PDT 2015	14	3C6FJU71TQSVCSSJSU4G5M069BCUYJ	A3NHMVYDRSEZBT	Approved	Fri Aug 21 14:51:32 PDT 2015	Fri Aug 21 15:19:30 PDT 2015	Mon Aug 24 15:19:30 PDT 2015	1678	Male	Yes	nice pay without extra work	0.03/image+0.02/per click	none	it may improve the likelihood of quality workers performing tasks	Yes	25	five	000040.jpg, 000430.jpg, 000518.jpg, 000528.jpg, 000712.jpg	10	No	people should be paid for their work	No	people should be paid for their work	No	I already tagged carefully
3FCO4VKOZ3CSLS22OJU9Z7ZM1LX7E2	Fri Aug 21 14:45:05 PDT 2015	14	3G5W44VEU7HHN5274AE8SZK42OPGKB	AXSDD92JCDPV7	Approved	Fri Aug 21 14:55:19 PDT 2015	Fri Aug 21 15:41:30 PDT 2015	Mon Aug 24 15:41:30 PDT 2015	2771	Male	Yes	Sounds reasonable	0.03/image+0.02/per click	I think it's pretty good the way it is. Maybe allow to create tags consisting of disjoint polygons.	I think it can increase its marketability and profitability.	Yes	42	5	000129.jpg, 000136.jpg, 000247.jpg, 000251.jpg, 000270.jpg	30	No		No	takes lots time without extra payment	Yes	It sounds like a great opportunity for everybody involved
3FCO4VKOZ3CSLS22OJU9Z7ZM1LX7E2	Fri Aug 21 14:45:05 PDT 2015	14	3OJSZ2ATDSVVCT634K68KXOT6Y2579	A22FJYYPL1KYI3	Approved	Fri Aug 21 19:33:29 PDT 2015	Fri Aug 21 20:00:07 PDT 2015	Mon Aug 24 20:00:07 PDT 2015	1598	Female	Yes	as stated above extra money is always good.	0.04/image+0.01/per click	suggest in the description of this HIT that you want workers to tag the color of each item. I didn't see anywhere that it said to put in the color and i am sorry that i didn't do it. I tried to	Could bring in more workers.	No	32	6	000950.jpg, 000900.jpg, 000902.jpg, 000904.jpg, 000914.jpg, 000934.jpg,	25	No	It was kind of fun tag a few, but tagging a lot needs more rewards	No	Wouldn't be worth it without a little extra reward	Yes	a thousand clicks in a day equals $10 for me. And more money is good.
3FCO4VKOZ3CSLS22OJU9Z7ZM1LX7E2	Fri Aug 21 14:45:05 PDT 2015	14	31LVTDXBL79JW1T35LKL63BQT1URLI	A1V13RCLTAE1X	Approved	Fri Aug 21 20:29:55 PDT 2015	Fri Aug 21 20:51:37 PDT 2015	Mon Aug 24 20:51:37 PDT 2015	1302	Female	Yes	I am always in favor of making bonus money and it does encourage people to tag more.	0.05/per click	I don't know of any currently.	I think it opens up a lot of potential sales and opportunities for workers who enjoy this type of work. It provides a service to the worker, the merchant, and the advertisement.	Yes	29	5	000696.jpg, 000701.jpg, 000708.jpg, 000712.jpg, 000739.jpg	6	No	I wouldn't mind doing several of these if there is extra pay. I think I would do a great job and be very quick and efficient.	No	Same as above.	Yes	This is something I would do
36QZ6V1588CVTMJG2M7PYGHLAPRSUI	Sat Aug 22 09:58:44 PDT 2015	10	34HJIJKLP5VFR03Z8DQELJYISP0V4I	A1D8CKTVHAVZ7I	Approved	Sat Aug 22 10:51:12 PDT 2015	Sat Aug 22 11:14:58 PDT 2015	Tue Aug 25 11:14:58 PDT 2015	1426	Female	Yes	at least this would result in possible extra pay	0.04/image+0.01/per click	it was pretty straight forward. no improvements needed	not sure	No	37	5	001885.jpg, 001902.jpg, 001914.jpg, 001923.jpg, 001960.jpg	23	No	would do more but would need pay	No	would do more but would need pay	Yes	at least this would result in possible extra pay
36QZ6V1588CVTMJG2M7PYGHLAPRSUI	Sat Aug 22 09:58:44 PDT 2015	10	34T446B1C0D2T3FYYV87JOKGQ510C3	A3L227HLQ3ASGK	Approved	Sat Aug 22 10:01:55 PDT 2015	Sat Aug 22 10:25:32 PDT 2015	Tue Aug 25 10:25:32 PDT 2015	1417	Male	Yes	you create a limitless bonus based on the work that you originally do. It justifies the work in the long run.	0.01/image+0.04/per click	personally I really like how the system is setup now. It's fast and effecient.	it opens up a better potential for workers.	Yes	34	5	001535.jpg, 01541.jpg 001571.jpg, 001630.jpg, 001669.jpg	19	No	I found it to be very fun and enjoyable, but time is money	No	work deserves rewards	Yes	I would focus more detail on the tagging yes as that would open up better results
36QZ6V1588CVTMJG2M7PYGHLAPRSUI	Sat Aug 22 09:58:44 PDT 2015	10	3DPNQGW4LLEUCAOY75I69RIJBWL46C	AN4P0SU1YSPF2	Approved	Sat Aug 22 12:53:22 PDT 2015	Sat Aug 22 13:12:16 PDT 2015	Tue Aug 25 13:12:16 PDT 2015	1134	Male	Yes	It goes to everything said above and appears to be a win-win-win situation (you-me-customer)	0.04/image+0.01/per click	As my first time, I did watch the video, but I would like to know more about the functionality/ability of the taking tools. So, I would be happy to do additional instructional videos.	There is on-going financial motivation for individuals to be involved in tagging. I would hope that preference is given to a select few workers for higher paid possibilities.	No	47	6	skirt, pants, purse, accessories, gloves, sunglasses	6	No	I'm honestly not sure what you mean by extra task rewards, but assuming their is not payment, then the answer is NO.	No		Yes	This makes sense to me as a way for both your company to profit on click-throughs and motivation for me to do it thoughtfully and well for my own benefit and so that the potential customers is seeing correctly identified material, etc.
36QZ6V1588CVTMJG2M7PYGHLAPRSUI	Sat Aug 22 09:58:44 PDT 2015	10	3FE7TXL1LIMDJESVI0DGMH2CRJQQ2O	A1PO6P10SESOQ9	Approved	Sat Aug 22 11:05:19 PDT 2015	Sat Aug 22 11:18:10 PDT 2015	Tue Aug 25 11:18:10 PDT 2015	771	Male	Yes	No comment, just the answer	$0.05/image	It's looks fine as it is	Hard to tell at the moment.. perhaps once implemented it'd be easier to say	Yes	34	5	002149, 002129, 002124, 002137, 002064	6	No	just did the minimum required to see what the task is all about...	No	a more detailed tagging can take lots of time	Yes	It's be intersting to see if the tagging technology allows it, though
36QZ6V1588CVTMJG2M7PYGHLAPRSUI	Sat Aug 22 09:58:44 PDT 2015	10	3FTF2T8WLRHT3V9GFIR1OTSCLPCW9J	AKP8JDWC6AXQ9	Approved	Sat Aug 22 10:14:09 PDT 2015	Sat Aug 22 10:50:13 PDT 2015	Tue Aug 25 10:50:13 PDT 2015	2164	Male	Yes	the extra bonus is important	0.04/image+0.01/per click	the tools need refinement to be more precise	beyond adding in-image advertisements functionality, I can not think of any other. I am new to tagging.	No	46	5	000366.jpg, 000054.jpg, 000715.jpg, 000028.jpg, 000701.jpg	9	No	it was relaxing. But I would need to find the time	No	Same as above.	Yes	it makes more sense now to do the work with potential extra rewards
36QZ6V1588CVTMJG2M7PYGHLAPRSUI	Sat Aug 22 09:58:44 PDT 2015	10	3L4PIM1GQTF3W002O09B0BDM2JIYR7	AHLEIW6IVQ98T	Approved	Sat Aug 22 10:10:42 PDT 2015	Sat Aug 22 10:45:41 PDT 2015	Tue Aug 25 10:45:41 PDT 2015	2099	Male	No	1 penny per tag is not worth it	$0.05/image	pay fairly!	I'm sure some people will make use of it, and some people will earn extra money with it. But it is not for me, thank you for the offer!	No	35	5	000347.jpg, 001765.jpg, 000430.jpg, 000433.jpg, 000451.jpg	8	No	I don't like working for free	No	my tagging was careful, and accurate enough	No	no guarantee my tags will be used, or clicked
36QZ6V1588CVTMJG2M7PYGHLAPRSUI	Sat Aug 22 09:58:44 PDT 2015	10	3LRLIPTPEQ8GDEP4IAMYWD9141YKA5	ARFGWITLWJ7IS	Approved	Sat Aug 22 12:23:24 PDT 2015	Sat Aug 22 12:33:07 PDT 2015	Tue Aug 25 12:33:07 PDT 2015	583	Female	Yes	{}	0.01/image+0.04/per click	following question 11, be able to chose the payment option you would like could be an interesting improvememt	It would make it more desirable because you can get paid even after doing the work	No	20	5	000712.jpg,000001.jpg,000950.jpg,000715.jpg,002080.jpg	5	No	I would want a higher reward to do more	No	{}	Yes	{}
36QZ6V1588CVTMJG2M7PYGHLAPRSUI	Sat Aug 22 09:58:44 PDT 2015	10	3N1FSUEFL5ZTXJ3DHSQKFXSBSIXD42	A1J66KNN9CTXXP	Approved	Sat Aug 22 11:06:50 PDT 2015	Sat Aug 22 11:54:52 PDT 2015	Tue Aug 25 11:54:52 PDT 2015	2882	Female	Yes	I would in the case I'd get paid every time my tag was clicked on an advertisement.	0.04/image+0.01/per click	I think you system is on track and doesn't need any improvement.	I think it's a great way to shop online.	Yes	49	6	000001.jpg, 000005.jpg, 000013.jpg,000028.jpg, 000040.jpg, 000054.jpg	12	No	I only like to tag images when I get paid for it	No	I only like to tag images when I get paid for it.	Yes	I would in the case I'd get paid every time my tag was clicked on an advertisement.
36QZ6V1588CVTMJG2M7PYGHLAPRSUI	Sat Aug 22 09:58:44 PDT 2015	10	3R8YZBNQ9HHK5FP1OCUBI8JK2Y97QG	A2Q1JUSJ9THUDZ	Approved	Sat Aug 22 10:29:28 PDT 2015	Sat Aug 22 10:57:00 PDT 2015	Tue Aug 25 10:57:00 PDT 2015	1652	Male	Yes	great idea!	0.04/image+0.01/per click	no improvements	it could potentially bring in more paying customers to specific items.	Yes	27	5	001856.jpg, 001861.jpg, 001814.jpg, 001868.jpg, 001876.jpg	17	Yes	I just enjoy trying new tasks.	No	it is hard to zoom in close enough.	Yes	tagging is awesome!
3FK4G712NWZFE9RCDHCPJFZXPTNSSL	Sat Aug 22 16:01:52 PDT 2015	10	33FBRBDW6OYKBSK1TZLMC3QY95FC88	A1H6U64BOALR2	Approved	Sat Aug 22 18:42:46 PDT 2015	Sat Aug 22 19:09:23 PDT 2015	Tue Aug 25 19:09:23 PDT 2015	1597	Female	No	It's such a low bonus and at the rate of maybe on the off chance someone clicks on it, not worth my time	0.04/image+0.01/per click	It is sort of a tedious task so I think giving task rewards would make the process more worth while, while the potential bonus will encourage quality of tagging	I'm not sure I understand what this question is asking, but I think the mechanics of this system run smoothly enough that I felt I could do this task easily and without trouble	No	22	5	002008.jpg, 002171.jpg, 002831.jpg, 001925.jpg, 000715.jpg	5	No	I don't feel like wasting my time	No	Because I already did it that way to begin with	No	Because like I said previously I already tagged everything I did as carefully as I could
3FK4G712NWZFE9RCDHCPJFZXPTNSSL	Sat Aug 22 16:01:52 PDT 2015	10	3E7TUJ2EGCLUUP9PYNMBW53REBHD96	AOSB3RSWQQENC	Approved	Sat Aug 22 21:15:06 PDT 2015	Sat Aug 22 22:01:42 PDT 2015	Tue Aug 25 22:01:42 PDT 2015	2796	Female	Yes	I work from home only.	$0.05/image	Already good.	Images load a bit slowly.	No	60	5	002124.jpg, 00 2129.jpg,002137.jpg,002162.jpg,002171.jpg	9	Yes	I like working!	Yes	I can improve my work.	Yes	I want to improve my accuracy.
3FK4G712NWZFE9RCDHCPJFZXPTNSSL	Sat Aug 22 16:01:52 PDT 2015	10	3I2PTA7R3TT80UWL76JKMDO68KBKQ5	A1823B3S2PJMGQ	Approved	Sat Aug 22 19:00:34 PDT 2015	Sat Aug 22 19:04:56 PDT 2015	Tue Aug 25 19:04:56 PDT 2015	262	Male	Yes	{}	0.01/image+0.04/per click	to select particular area of the image	it's pretty good and informative	Yes	32	2	000008.jpeg, 000007.jpeg	8	No	{}	No	{}	Yes	{}
3FK4G712NWZFE9RCDHCPJFZXPTNSSL	Sat Aug 22 16:01:52 PDT 2015	10	3I33IC7ZWF1LWYLET4GUWHN7N282AZ	ADS7SG188006Z	Approved	Sat Aug 22 20:42:28 PDT 2015	Sat Aug 22 21:08:54 PDT 2015	Tue Aug 25 21:08:54 PDT 2015	1586	Female	Yes	{}	0.04/image+0.01/per click	have people tage every thing in the immage instead of leaving part of the immage untaged	was a good start for the image tagging start on mturk	Yes	23	5	{}	5	No	Becuase I would not do more because I need the money	Yes	I try to do hight quality tagging no matter what	No	becuae I did tag very carefully
3FK4G712NWZFE9RCDHCPJFZXPTNSSL	Sat Aug 22 16:01:52 PDT 2015	10	3JV9LGBJWTD0DWNM2261ZG1DM12OGM	ATSRKL3EWQ8DU	Approved	Sat Aug 22 16:31:50 PDT 2015	Sat Aug 22 16:52:58 PDT 2015	Tue Aug 25 16:52:58 PDT 2015	1268	Female	Yes	{}	0.01/image+0.04/per click	none, it was a simple task	It will improve it. most people don't know about tagging. Having these tasks brings awareness and a means of profitability	No	48	5	000001.jpg added blue shade, 000013.jpg- added blue tag to shoes,000005.jpg added red to her hair, 000934.jpg did a few tags on this one, 000542.jpg	9	No	There must be an award to the task	No	I don't have a practical use for it	Yes	{}
3FK4G712NWZFE9RCDHCPJFZXPTNSSL	Sat Aug 22 16:01:52 PDT 2015	10	3NG53N1RLVI35HT5Y47SW3P58NN8P9	A2K2BS73NCFEDU	Approved	Sat Aug 22 17:00:28 PDT 2015	Sat Aug 22 17:41:30 PDT 2015	Tue Aug 25 17:41:30 PDT 2015	2462	Female	Yes	{}	0.05/per click	More images to tag	I enjoyed this hit and would be interested in more	Yes	23	5	002199.jpg, 002235.jpg, 002281.jpg, 002305.jpg, 002307.jpg	9	No	I would require a reward	No	I would like a reward	Yes	Better reward
3FK4G712NWZFE9RCDHCPJFZXPTNSSL	Sat Aug 22 16:01:52 PDT 2015	10	3PQMUDRV7R54D1IEUVTZ0VI7P5UIIC	A2C0F7CXEEHBRB	Approved	Sat Aug 22 16:11:44 PDT 2015	Sat Aug 22 16:38:13 PDT 2015	Tue Aug 25 16:38:13 PDT 2015	1589	Female	Yes	I am being paid for my work at a reasonable rate	0.04/image+0.01/per click	faster image download times	make such tasks easier	Yes	29	6	001765.jpg, 000430.jpg, 000433.jpg, 000040.jpg, 000739.jpg, 002064.jpg	7	No	too time consuming with out additional benefits	No	too time consuming without additional benefits	Yes	I am being paid for my work at a reasonable rate.
3FK4G712NWZFE9RCDHCPJFZXPTNSSL	Sat Aug 22 16:01:52 PDT 2015	10	3VD82FOHKQNNWK7H0GAI9PGYO9BCO1	A9KUILSN9FEKP	Approved	Sat Aug 22 16:48:14 PDT 2015	Sat Aug 22 17:09:46 PDT 2015	Tue Aug 25 17:09:46 PDT 2015	1292	Female	Yes	we should still get paid more for better work, but this would be a nice bonus	0.04/image+0.01/per click	It would be nice if there was a way to avoid the person's arms when doing the tagging since it makes it hard to connect some of the polygons. It would also be nice to have a rounded feature to go around necklines.	I think more people will be willing to work your tasks if there's an incentive, but most people will not work for such low pay	Yes	27	5	002093.jpg, 002095.jpg, 002100.jpg, 002120.jpg, 002115.jpg	16	No	I want to earn more money	No	if I'm doing higher quality work, I should get paid more	Yes	if there's a chance of higher payment, of course
3FK4G712NWZFE9RCDHCPJFZXPTNSSL	Sat Aug 22 16:01:52 PDT 2015	10	3VE8AYVF8MWRE44BK1BNK9B16HYF8B	AVQJTOVGFSLAT	Approved	Sat Aug 22 16:23:27 PDT 2015	Sat Aug 22 17:01:43 PDT 2015	Tue Aug 25 17:01:43 PDT 2015	2296	Female	Yes	Same answer, more money is always a good thing, even if it is only a possibility.	0.03/image+0.02/per click	The pictures took a pretty long time to load. I think I could get more tagging done if it didn't take so long. I have really high speed internet so it shouldn't take that long.	None. I think it would just irritate the workers. But that's just my opinion.	Yes	34	7	002067.jpg, 002035.jpg, 002039.jpg, 002058.jpg, 002061.jpg, 002064.jpg, 002065.jpg	44	Yes	This was actually pretty fun.	No	That seems like a little too much work for only .50 cents.	Yes	Because more money is always a good thing.

total participants	35				
female	18				
male	17				
total tagged image per person	5.114285714				
tag more without extra pay? =Yes	3		tag more with image-click-ads =Yes	33	
tag more carefully without extra pay? =Yes	2		tag more carefully with image-click-ads? =Yes	25	8/10 think they've tagged very carefully, 1/10 thinks not necessary to get very accurate region, 1/10 thinks no guarantee for clicks

$0.05/image	$0.04/image+$0.01/click	$0.03/image+$0.02/click	$0.02/image+$0.03/click	$0.01/image+$0.04/click	$0.05/click
5	13	4	4	5	4



Original cost	current cost	improvement
1.75	1.02	0.4171429