CO-FILTERING HUMAN INTERACTION AND OBJECT SEGMENTATION

A Degree Thesis

Submitted to the Faculty of the

Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona

Universitat Politècnica de Catalunya

by

Ferran Cabezas Castellví

In partial fulfilment

of the requirements for the degree in

AUDIOVISUAL SYSTEMS ENGINEERING

Advisors: Axel Carlier, Vincent Charvillat, Xavier Giró-i-Nieto and Amaia Salvador .

Barcelona, February 2015

Abstract

This thesis explores processing techniques to deal with noisy data in crowdsourced object segmentation tasks. It is used the data collected with Click'n'Cut, an online interactive segmentation tool, and it is performed several experiments towards improving the segmentation results.

First, it is introduced different superpixel-based techniques to filter users' traces, and assess their impact in the segmentation result.

Second, it is presented different criteria to detect and discard the traces from potential bad users, resulting in a remarkable increase in performance.

Then, it is shown a novel superpixel-based segmentation algorithm which does not require any prior filtering and is based on weighting each user's contribution according to his/her level of expertise.

Finally, it is exposed different features and their corresponding rules for automatic categorizing the crowd users. The main application of this automatic categorization is to observe which pattern follow each user that produces bad traces and convert these traces into traces that give us better performance in the segmentation

Resum

Aquesta tesi explora les tècniques de processament per fer front a les dades amb soroll en les tasques de segmentació d'objectes mitjantçant l'interacció humana a gran escala. S'utilitza la informació recollida amb l'interficie Click'n'Cut, una eina web de segmentació interactiva on es porten a terme diversos experiments per a la millora dels resultats de la segmentació.

En primer lloc, s'introdueixen diferents tècniques basades en superpixels per filtrar els marcadors fets pels usuaris, i després s'avalua el seu impacte en el resultat de la segmentació.

En segon lloc, es presenten diferents criteris per detectar i descartar els marcados de possibles mals usuaris.

A continuació, es mostra un nou algorisme de segmentació basat en superpíxels que no requereix cap tipus de filtrat previ i es basa en la ponderació de la contribució de cada usuari d'acord al seu nivell d'experiència.

Per últim, s'exposen diferents característiques i les seves regles de corresponents per l'automàtica categorització dels usuaris capturats en la campanya microworkers. La principal aplicació d'aquesta categorització automàtica és observar quin patró segueix cada usuari que produeix marcadors incorrectes i intentar reconvertir aquests marcadors en d'altres que incrementin els resultats de la segmentació.

Resumen

Esta tesis explora las técnicas de procesamiento para hacer frente a los datos con ruido en las tareas de segmentación de objeto a grande escala. Se utilizan los datos recogidos con Click'n'Cut, una herramienta web de segmentación interactiva, y se llevan a cabo varios experimentos para la mejora de los resultados de la segmentación.

En primer lugar, se introduce diferentes técnicas basadas en superpixels para filtrar los marcadores de los usuarios, y evaluar su impacto en el resultado de la segmentación.

En segundo lugar, se presenta diferentes criterios para detectar y descartar los marcadores de posibles malos usuarios.

A continuación, se muestra un nuevo algoritmo de segmentación basado en superpixels que no requiere ningún tipo de filtrado previo y se basa en la ponderación de la contribución de cada usuario de acuerdo a su nivel de experiencia.

Por último, veremos las diferentes características y sus correspondientes reglas para la automática categorización de los usuarios capturados en la campaña microworkers. La principal aplicación de esta categorización automática es observar qué patrón sigue cada usuario que produce malos marcadores e intentar reconvertir estos marcadores en otros que incrementen el rendimiento de la segmentación.

Acknowledgements

Foremost, I would like to express my sincere gratitude to my advisor Xavier Giró for accepting me to work with him and making possible to go abroad to realise my final degree thesis. Furthermore, I appreciate his contributions and dedication on the follow-up of the whole thesis.

My sincere thank also go for Prof. Vincent Charvillat for welcoming me. It was an honour to work in his laboratory. Besides, I appreciate all contributions he made.

Special thanks go to my advisor Axel Carlier, he helped me every time I had a problem. I also appreciate all his time spent on my thesis follow-up and all new perspectives he proposed to realize my thesis fuller.

Finally, I would like to thank Amaia Salvador. Although she joined later, she has made essential contributions in the final thesis.

Revision history and approval record

Revision	Date	Purpose
0	16/02/2015	Document creation
1	19/02/2015	Document revision
2	22/02/2015	End of document

DOCUMENT DISTRIBUTION LIST

Name	e-mail
Ferran Cabezas Castellvi	fcab65@gmail.com
Xavier Giró	xavier.giro@upc.edu
Vincent Charvillat	vcharvillat@gmail.com
Axel Carlier	carlier.axel@gmail.com
Amaia Salvador	amaia91@gmail.com

Written by:		Reviewed and approved by:	
Date	16/02/2015	Date	19/02/2015
Name	Ferran Cabezas	Name	Xavier Giró
Position	Project Author	Position	Project Supervisor

Table of contents

The table of contents must be detailed. Each chapter and main section in the thesis must be listed in the “Table of Contents” and each must be given a page number for the location of a particular text.

List of Figures

List of Tables:

1. Introduction

Object segmentation is one of the most challenging problem that is still present in computer vision. It consists in, for a given object in an image, assigning to every pixel a binary value: 0 if the pixel is not part of the object, and 1 otherwise. In particular, this project is focused on interactive object segmentation, that is, object segmentation assisted by human feedback, that, in our case we have used the information of humans from a crowdsourcing campaign.

1.1. Goal

The main purpose of this project is to investigate the problematic of visual content analysis. As it has been said, the work is focused on image segmentation using the analysis of how humans interact with the visual content.

However, information that humans provide about the visual content is not always reliable. For this reason, it is wanted to maintain just the useful information provided by each human.

Therefore, being more concrete, we will work on techniques for filtering the user interaction that have been captured in order to obtain a better object segmentation in an image. Figure 1 shows an example of a good and a bad user interaction and their result.

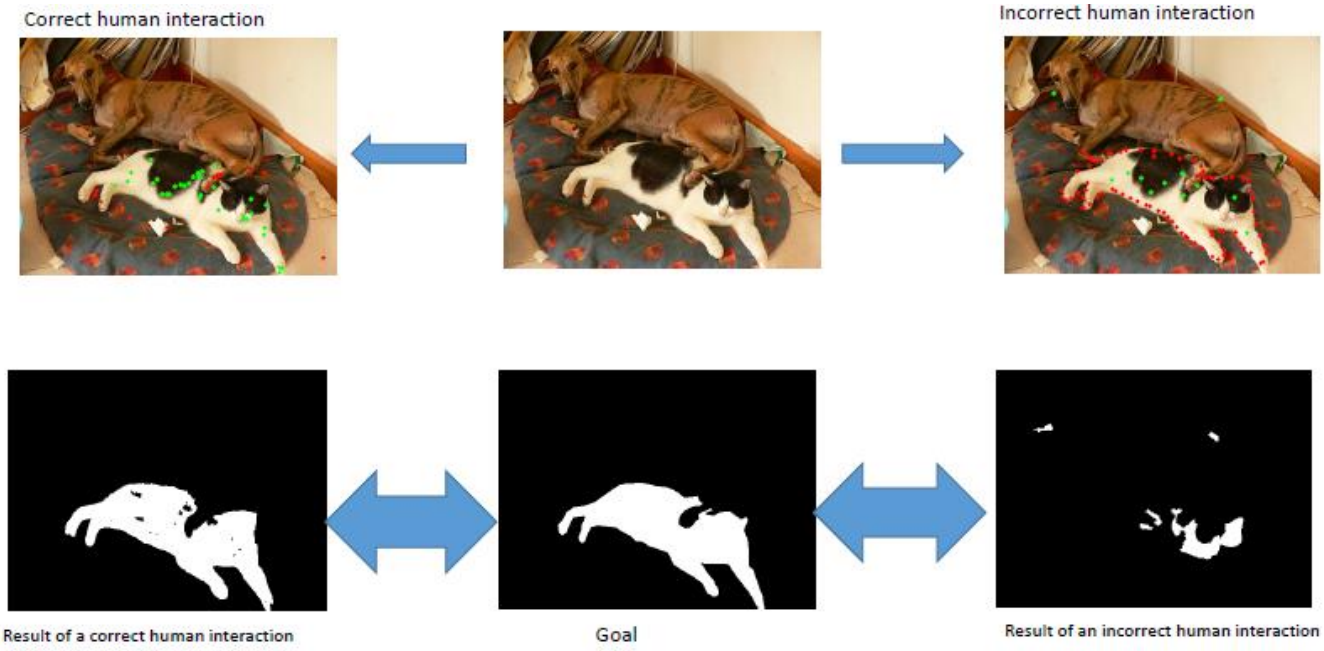

Figure 1: Result of a good and a bad human interaction

1.2. Overview

This project is an extension of the doctorate thesis already finished called 'Combining Content Analysis with Usage Analysis to better understand visual contents'[24] of the author Axel Carlier, who is also one of the project supervisors.

This project is also based on the following papers: 'Click'n'Cut: Crowdsourced Interactive Segmentation with Object Candidates'[3] and 'Crowdsourced object segmentation with a game'[4], whose authors are: Axel Carlier, Amaia Salvador, Xavier Giró-i-Nieto, Oge Marques and Vincent Charvillat. Besides, I have received the suitable tools, code and data for the correct realization of the project.

My project supervisors provided me at the starting of the work, a list of interesting ideas to focus my work on. Therefore, as my priority have always been working on an image processing project, both, my supervisors and me, agreed to work on a project that could motivate us during its whole realization.

1.3. Project Requirements and specifications

Project requirements:

- Advanced Matlab skills for understand and develop.
- Medium knowledge on image processing
- Advanced capacity of researching scientific papers to exploit their content.
- Advanced level of English for the oral and written communication
- Basic level of French for the oral communication

Project specifications:

- The whole project has been designed and coded in Matlab.
- Using the same train and test set from the state of the art so as can be compared the results.

1.4. Work plan

The realization of this Final Degree Thesis has followed the initial work-plan set, in exception of some incidences that will be commented on chapter 1.5 of this document.

1.4.1. Work packages

WP 1: Project documentation

WP 2: Set-up an evaluation system

WP 3: Filtering based on oversegmentation

WP 4: Filtering based on users

WP 5: Combining oversegmentation and users filtering.

WP 6: Natural language processing

1.4.2. Gant diagram

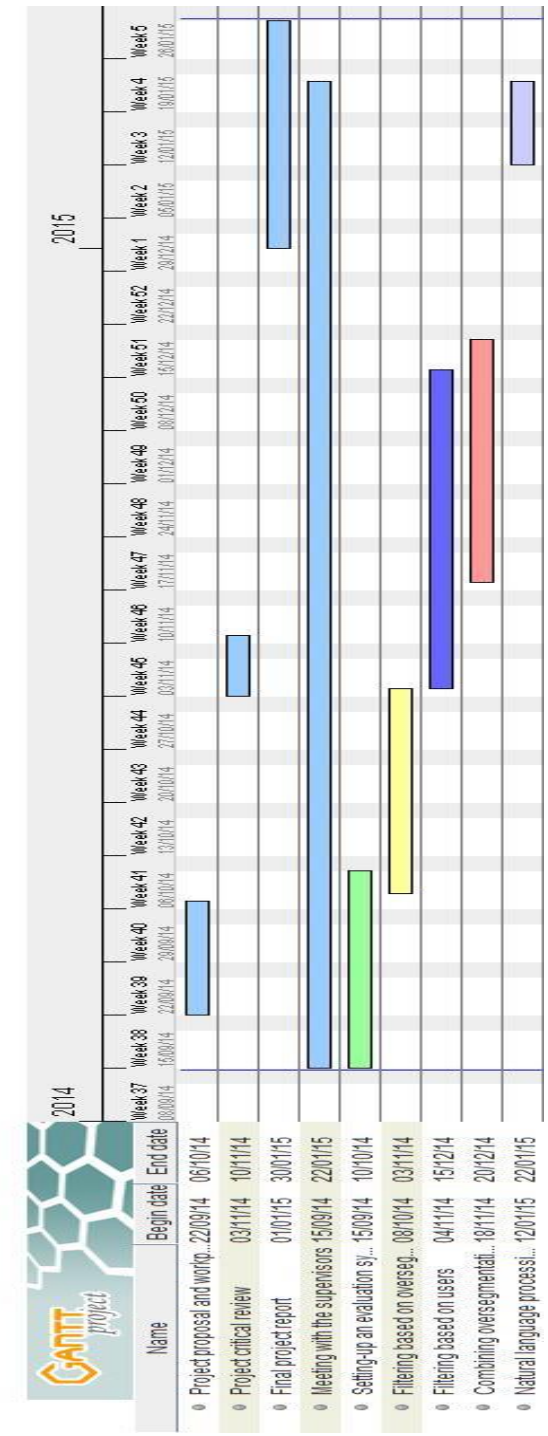

Figure 2: Gant diagram

1.5. Incidences and modifications

The whole work has roughly followed the planning set on 1.4.1 and 1.4.2. However, there have been made some slight modifications from the Critical review document.

First of all, and just for a better understanding, it is considered the WP 3 called 'Filtering based on over segmentation' as a 'Filtering based on clicks'. Consequently, WP 5 it is called 'Filtering clicks and users'.

It is introduced two new work packages, WP7 and WP8, called 'Foreground map algorithm' and 'Automatic user categorization'. These two tasks have been purchased after finishing the WP5. In chapter 3 will be explained in more detail what these work packages are about.

Because of the lack of time, the WP 6, called 'Natural language processing', has been omitted since we focused on first performing WP7 and WP8 as they produce more related results on WP 2-5 compared to that one.

Finally, the realization of this document, 'Final project report' in the Gant diagram, has not followed the work-plan as it has been done almost in out of time because of a misunderstanding between my advisors and the academic secretary.

2. State of the art:

The combination of image processing with human interaction has been extensively explored in the literature. Many works related to object segmentation have shown that user inputs throughout a series of weak annotations can be used to either seed segmentation algorithms or to directly produce accurate object segmentations. Researchers have introduced different ways for users to provide annotation for interactive segmentation: by drafting the contour of the objects [1, 2], generating clicks [3, 4, 5] or scribbles [6, 7] over foreground and background pixels, or growing regions with the mouse wheel [8].

However, the performance of all these approaches directly relies on the quality of the traces that user's produce, which raises the need of robust techniques for Quality Control of human traces. The authors in [9] add gold-standard images in the workflow with a known ground truth to classify users between"scammers", users who do not understand the task and users who just make random mistakes. In [2], users are

discarded or accepted based on their performance in an initial training task and are periodically verified during the whole annotation process. In any case, authors in [10] have demonstrated the need for tutorials by comparing the performance of trained and non-trained users. Quality control can also be a direct part of the experiment design. The Find-Fix-Verify design pattern for crowdsourcing experiments was used in [11] for object detection by defining three user roles: a first set of users drew bounding boxes around objects, others verified the quality of the boxes, and a last group of checked whether all objects were detected. Luis Von Ahn also formalized several methods for controlling quality of traces collected from GamesWith A Purpose (GWAP) [12]. In the following pictures, [22] and [23], can be appreciated two interfaces based on GWAP.

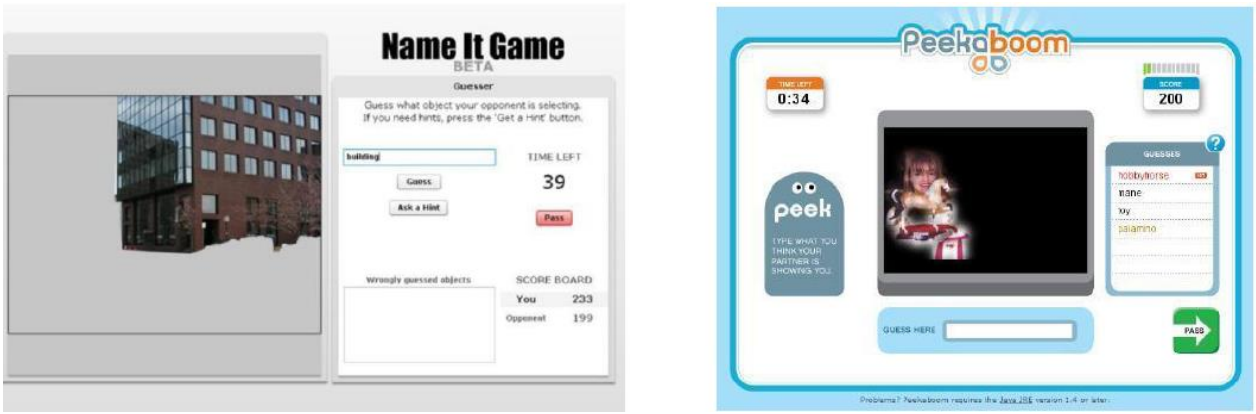

Figure 3: Example of Games with a Purpose interfaces

Quality control can also introduced at the end of the study as in [13], where a task-specific observation allowed discarding users whose interaction patterns were unreliable. Quality control may not exclusively focused in users but also on the individual traces, as in [14, 15]. One option to process noisy traces is collecting annotations from different workers and compute a solution by consensus, such as the bounding boxes for object detection computed in [16].

2.1. Click'n'Cut

As it has been exposed in chapter 1.2, this work is focused on the interactive object segmentation web tool called Click'n'Cut. Figure 4 shows a screenshot of the Click'nCut interface. The interface consists on displaying the image that we wish to segment, along with a set of basic interactions (on the bottom-right of the screen) and a reminder of how the interface works (on the top-right part of the screen). There is also a description of the

object to segment on the top of the screen, right above the image. On the figure the object to be segmented is the cat.

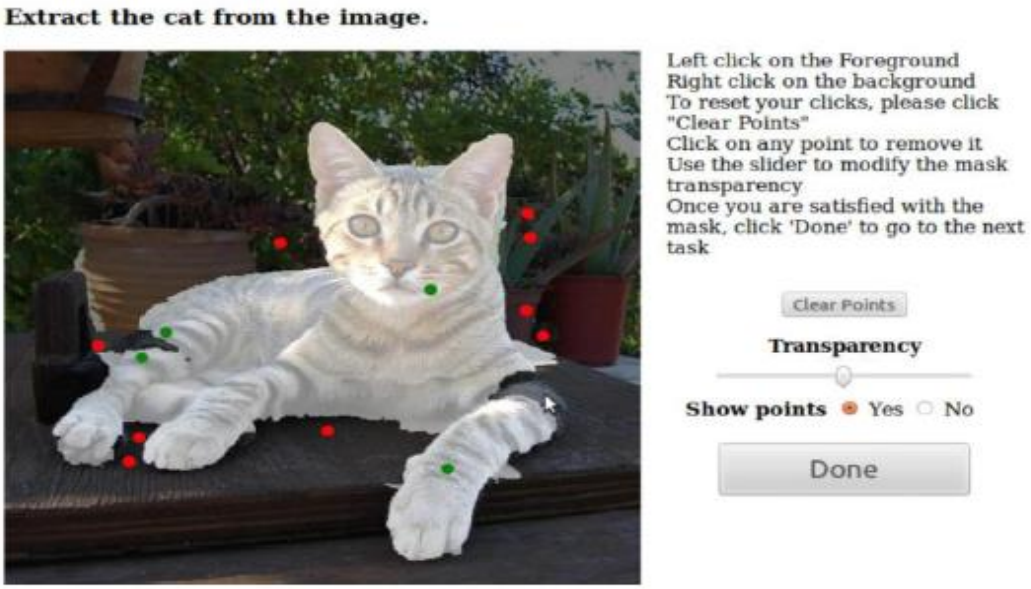

Figure 4: Click'n'Cut interface

The fundamental interactions available to users are the left and right clicks. A left click on the image indicates a *foreground* point (in green) whereas a right click on the image indicates a *background* point (in red). After each click the current version of the segmentation is updated and displayed over the image with an alpha value of 0.5 by default. At any time the user can choose to modify the alpha using the *Transparency* slider to either get a better look at the image or to better see the current mask.

A user can also remove a bad click: just clicking on it again makes it disappear. The *Clear* points button removes the entire set of clicks that have been made by the worker. Finally, once satisfied with the result, the user can go on to the next task by clicking the *Done* button.

The user can also choose not to display the points (annotations), in order to have a clearer view of the current state of the segmentation. The radio buttons "Show points" serve this purpose.

2.1.1. Data acquisition

In our work we used the data collected by [3] over two datasets:

- 96 images, associated to 100 segmentation tasks, are taken from the DCU dataset [7], a subset of segmented objects from the Berkeley Segmentation

Database [17]. These images will be referred in the rest of the paper as our test set.

- 5 images are taken from the PASCAL VOC dataset [18]. We use these images as gold standard, i.e. we use the ground truth of these images to determine workers' errors. These images form our training set.

20 users performed the entire set of 105 tasks from a crowdsourcing campaign called microworkers.com.

2.1.2. Obtaining the masks from the clicks

Once we have the clicks from all users exist two different techniques for segmenting the object (obtaining the mask):

2.1.2.1. Combination of precomputed binary object candidates

The first technique [3] is based on the combination of different precomputed MCG binary object candidates [19] according to their correspondence to the user's clicks. For example, in figure 5, the mask from the top-right is the best mask with respect to the two clicks (foreground in green and background in red) since it is the only one that is consistent with the two clicks. If the user was to label a pixel o the cat's head as foreground, then no mask would be consistent with the three clicks. The best mask would therefore be the combination of the three first masks which would all be consistent with the background click and at least one foreground click. The fourth mask on the figure is clearly identified as too big since it contains a background click.

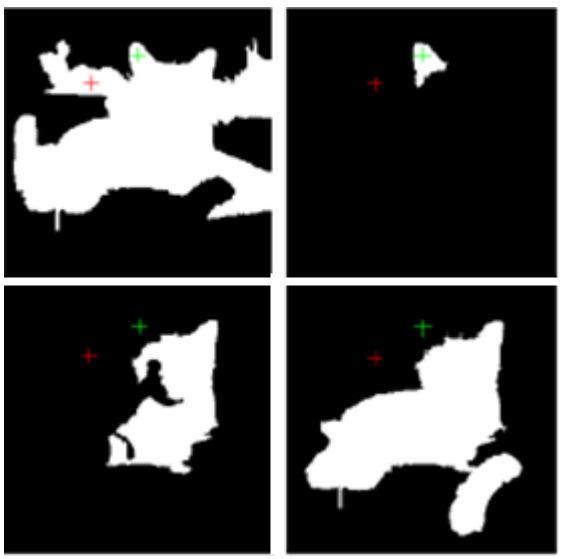

Figure 5: Precomputed object candidates

2.1.2.2. Foreground map algorithm

The other technique for obtaining the mask given all clicks it is called foreground map algorithm [24]. In figure 6 it is shown how this algorithm works: Given the set of clicks from all users, each superpixel is labelled with a number between 0 (background) and 1 (foreground). To compute these labels it is leveraged each worker contribution by a measure of the worker's confidence, based on this worker's performance on the gold standard images (train set). For example, if a worker *w* has a 5% error rate on the gold standard images, the measure of confidence c_w for this worker will be 0.95. A foreground (resp.background) click brings a contribution to the superpixel of c_w (resp 1- c_w).

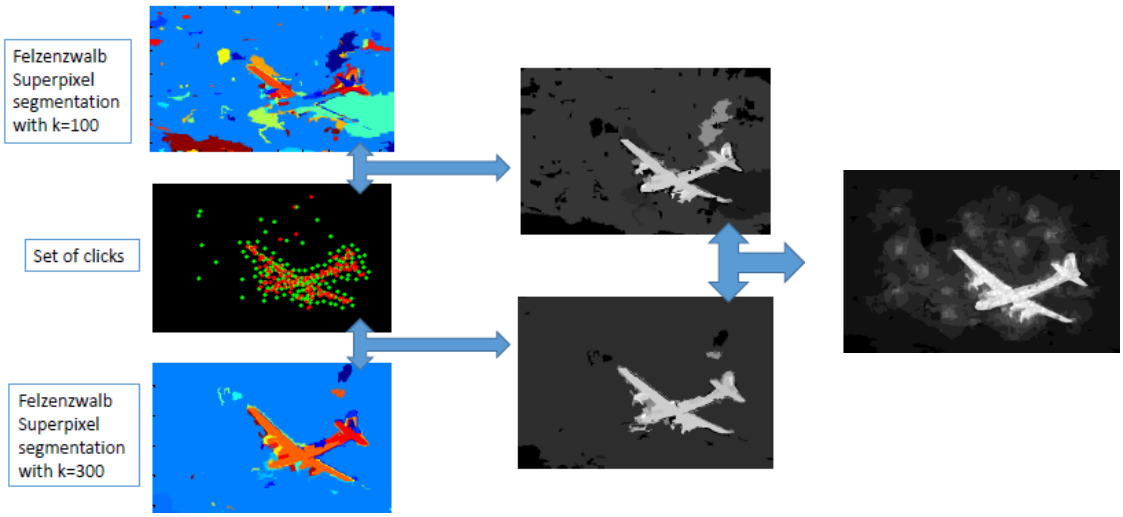

Figure 6: Steps for the formation of the foreground map

To limit the the influence of the superpixel segmentation, it is performed the computation on several different superpixels segmentations and average the respective results. In figure 6 it is used Felzenszwalb algorithm [21] with different parameters (k=100 and k=300) to obtain the final foreground map. By thresholding and applying a simple hole filling to this foreground map it is obtained the final mask. The advantage of this technique is that all clicks are used. This is an important consideration that will be explained deeper in the following sections.

2.1.3. Evaluation of the results

Once it is obtained the mask, either by applying the combination of object candidates or the foreground map algorithm, it is compared with its correspondence Ground truth mask.

The measure used is the Jaccard index. Therefore, given two different sub-spaces *A* and *B*, in our case are the Ground truth mask and the predicted mask, it is computed:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}.$$

A Jaccard index value close to 1 mean a good similarity and a value close to 0 mean a bad similarity between both sub-spaces.

2.1.4. Previous results

On the test set, experiments on expert users recruited from computer vision research groups reached an average Jaccard of 0.93 with the best algorithm in [7]. On the other hand, a value 0.89 was obtained with the same Click'n'Cut [3] tool used in this work, but on a different group of expert users. However, the group of crowdsourced workers performed significantly worse with Click'n'Cut, with a result of 0.14 with raw traces, which increased up to 0.83 when filtering worst performing users. In following sections it is proposed more sophisticated filtering techniques to improve this values.

3. Processing of human interaction

In this section it is exposed different techniques for treating with human interaction. The first sub-section it is focused on detecting and removing bad interactions. This filtered data is used to feed the object segmentation algorithm commented in section 2.1.2.1 and presented on [3]. On the second sub-section it is proposed an improvement of the foreground map algorithm. And, since it has already been exposed in section 2.1.2.2, this technique is an alternative to not remove any human interaction since all clicks are used to create the foreground map.

3.1. Removing human interaction

In figure 7 can be seen the impact of all users in the final mask and proposes us the necessity of discarding bad human interactions. Moreover, in figure 7 can be seen that by just removing 'bad users' the resulting mask improves. This propose us that a lot of errors can be removed just by discarding bad users.

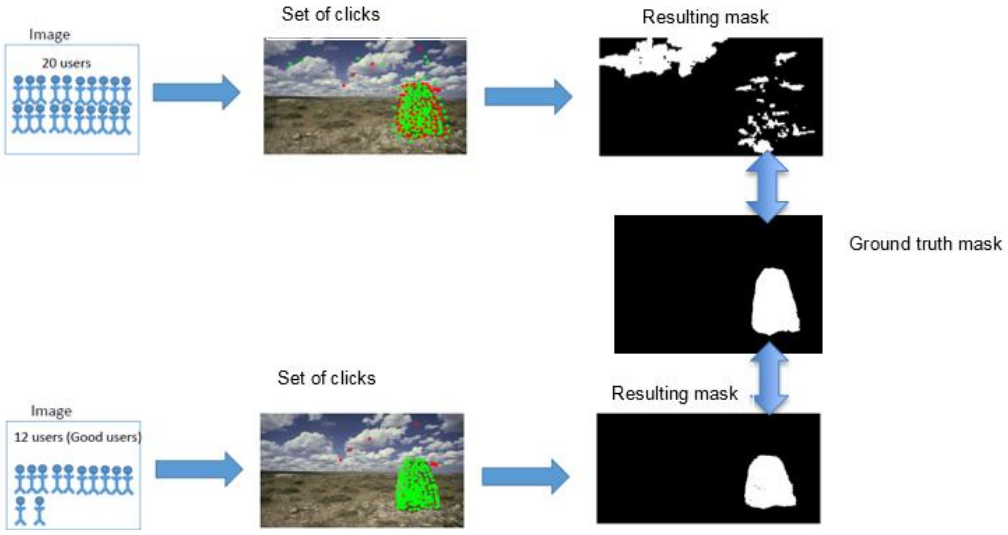

Figure 7: Impact of good and bad users to the resulting mask

3.1.1 Removing users

In this section it is proposed to use our training set as a gold standard to determine which users should be ignored. In particular, it will be used two different features to separate good from bad users: their error rate and their average Jaccard index. Figure 8 depicts this concept: for each user it is calculated their Jaccard index or error rate based on the 5 gold standard images and it is removed users based on an error rate or Jaccard index threshold

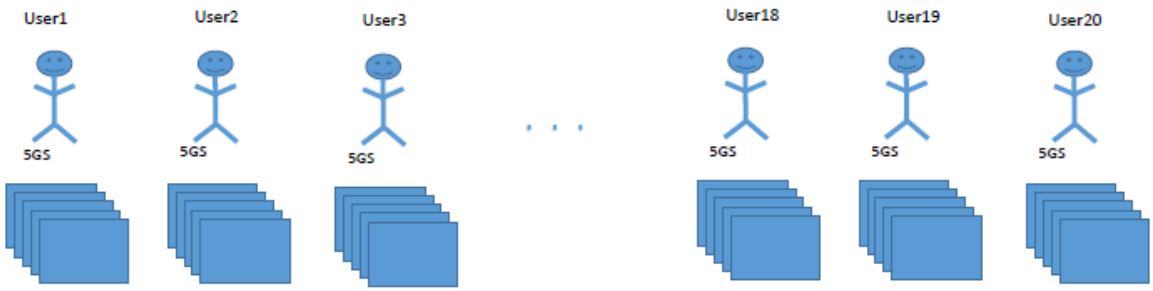

Figure 8: Visualization of how are calculated error rate and Jaccard index for each user in the train set

When a user is considered as bad, it is removed all its contribution in the final result. Despite bad users' tend to generate bad traces, not all bad users produce bad traces in

all images. Therefore, in the next section will be focused on removing bad clicks so as for being more consistent in the final result.

3.1.2 Removing clicks

Given an image with all traces from all users, all clicks are processed and are removed just the clicks considered as a 'wrong clicks'. Figure 9, shows this idea.

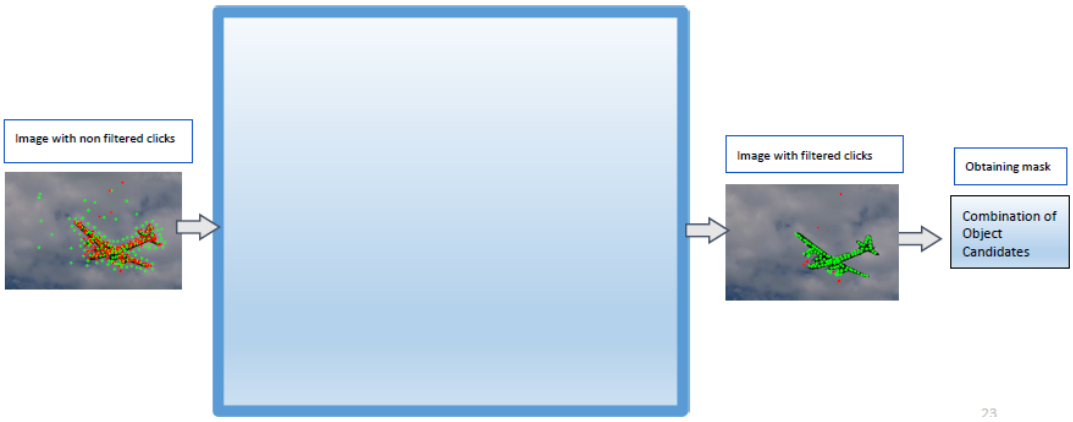

Figure 9: Schematic of the click removal

Wrong clicks can be detected by looking at other clicks in its spatial neighbourhood. Nevertheless, spatial proximity is not enough because the complexity of the object may actually require clicks from different labels to be close, especially near boundaries and salient contours. For this reason, the first step is to over segment the image. In particular, it is used SLIC [20] and Felzenszwalb [21] superpixel techniques. On figure 10, it is shown the 6 possible click distributions that can occur given a superpixel: higher number of foreground than background clicks, higher number of background than foreground clicks, same number of background and foreground clicks, foreground clicks only, background clicks only and no clicks.

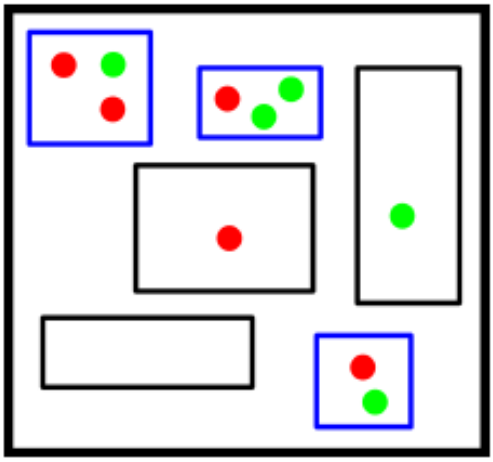

Figure 10: Possible configurations of background (in red) and foreground (in green) clicks inside a superpixel. Superpixels containing conflicts are represented in blue.

Among these six configurations, the three first ones reveal conflicts between clicks. At this point, it is needed to find some algorithm to remove these clicks that are in conflicting superpixels. Figure 11 depicts the two different methods that have been considered to solve the conflicts: keep only those clicks which are majority within the superpixel (left), or discard all conflicting clicks (right).

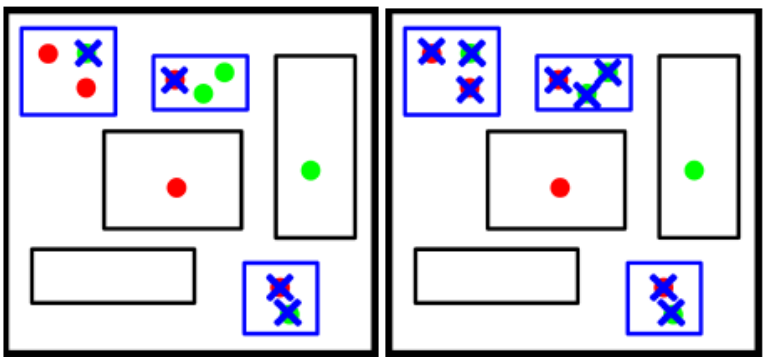

Figure 11: Two options to solve conflicts: keep majorities(on the left) and discard all(right)

3.2. Improving Foreground map

Section 2.1.2.2 has introduced the foreground map by just using Felzenszwalb [21] superpixel technique. In order to obtain better results, the same experiment was repeated by introducing SLIC [20] superpixel technique with different parameters and adding as well, more parameters to Felzenszwalb. In particular, the experiment was run by taking the following parameters *k* from Felzenszwalb [21]: 10, 20, 50,100,200,300,400 and 500. With SLIC [20] it is considered different values of region size 5, 10,20,30,40 and 50. For each parameter of the different superpixels technique it is obtained a different foreground map. Therefore, by combining all these foreground map and normalizing the values of the superpixels between 0 and 1 it is obtained the foreground maps of the figure 12.

Figure 12: Resulting foreground maps

The last step in order to obtain the binary mask is to set a threshold value and apply a simple hole filling algorithm in the resulting foreground map

4. Automatic categorization of the users

Previous section have presented how to treat with human interactions, either by removing bad human interaction or by giving a measure of confidence to each click without removing any information. This section will be focused on automatically categorizing the users and trying to convert bad human interactions into good ones. For this reason, the first step will be to see which pattern follow each user in all images. Once a user is categorized, it is easier to decide which conversion can be applied to that user in order to obtain a better human interaction. In the following chapter will be presented the different categories of the users that have been found.

4.1. User categories

- **Painter**: Lot of foreground clicks inside the object to segment

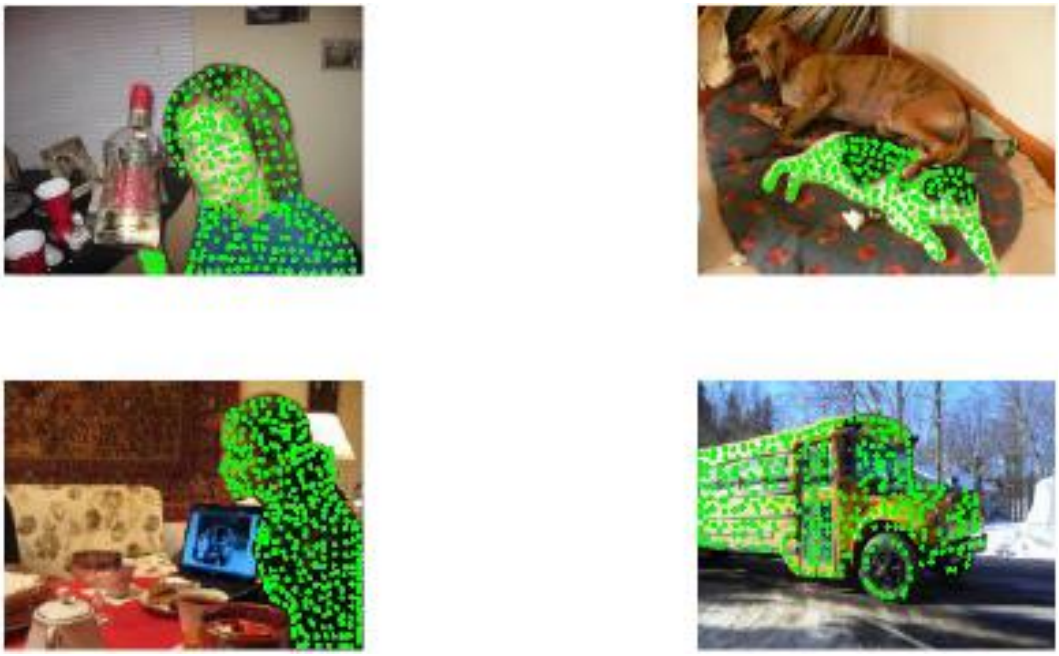

Figure 13: Painter user

- **Tired**: Few clicks per image

Figure 14: Tired user

- **Border guard**: Most of the background clicks are in the contour of the image.

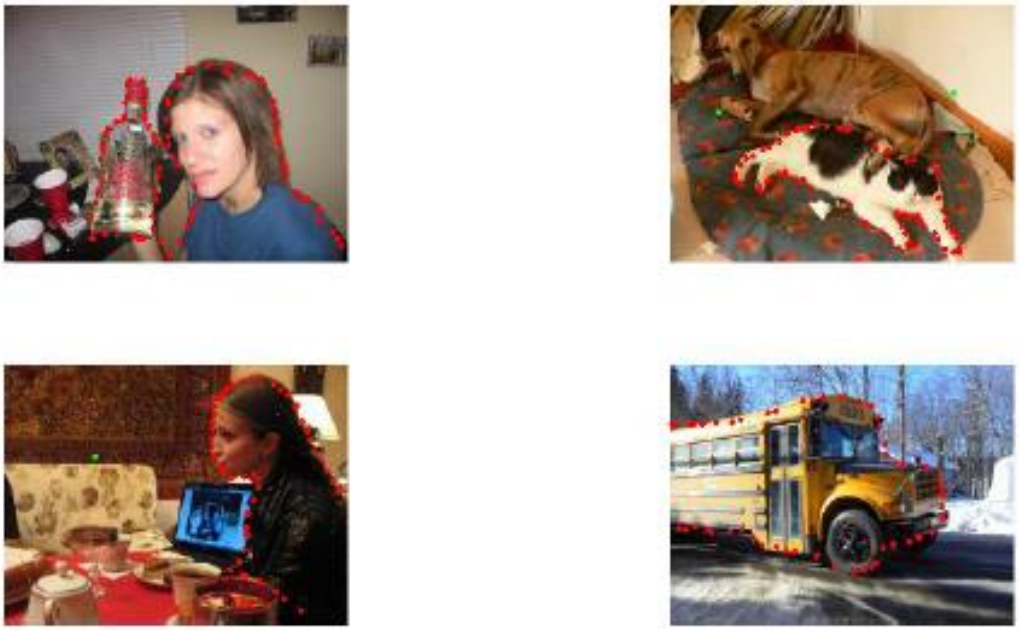

Figure 15: Border guard user

- **Surrounder**: Most of the foreground clicks are in the contour of the image.

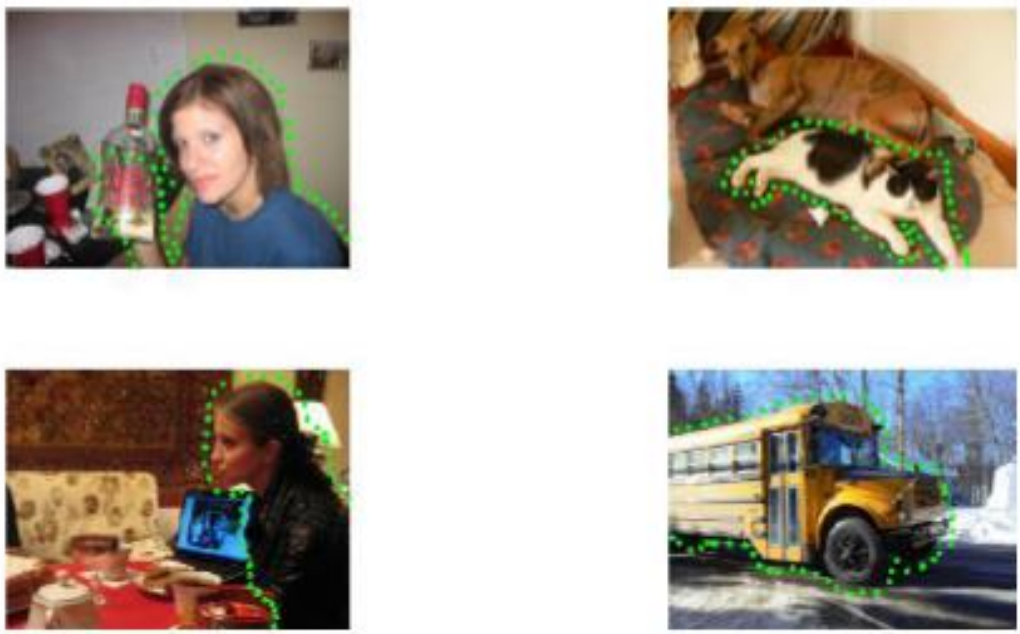

Figure 16: Surrounder user

- **Mirror**: Have understood the experiment upside-down

Figure 17: Mirror user

- **Spammer**: Has randomly placed foreground clicks over the image

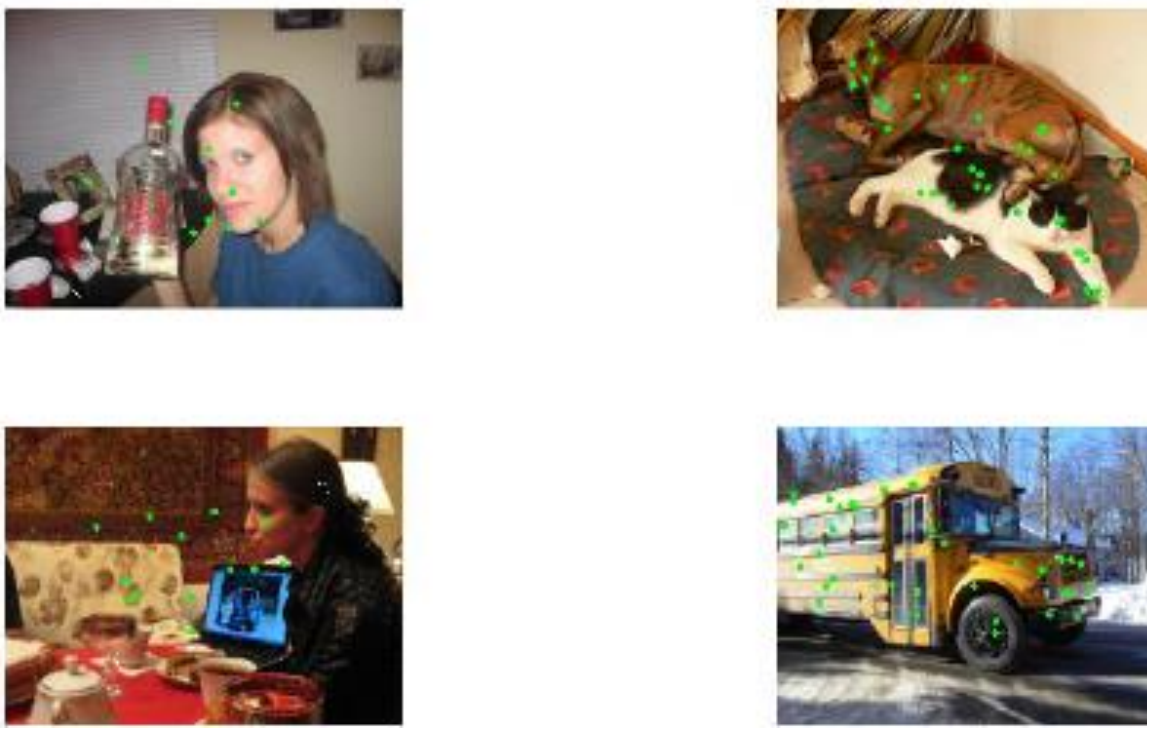

Figure 18: Spammer user

- **Experts**: Have well-understood the experiment and just made few mistakes

Figure 19: Expert user

- **Different pattern**: Does not follow the same pattern of clicks in all images.

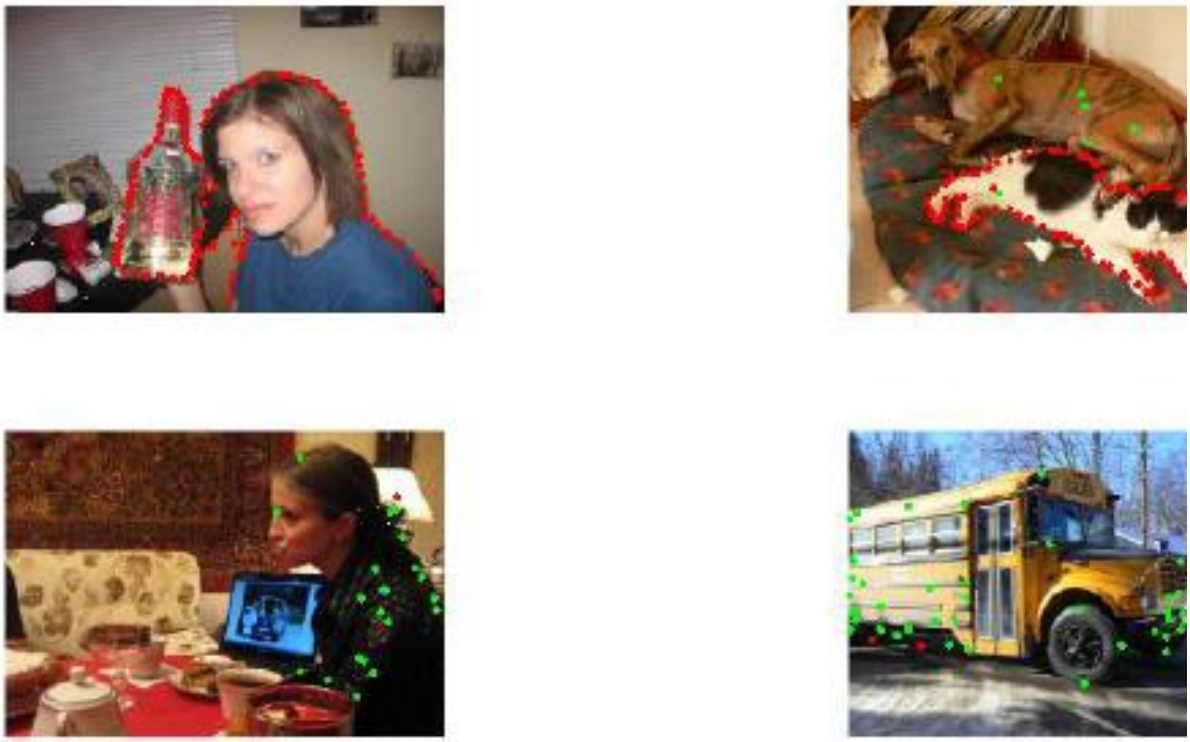

Figure 20: Different pattern user

A good example of bad human interaction conversion could be in the case of the mirror user, by just exchanging foreground for background clicks. Or, in the case of border guard and surrounder users, instead of removing their clicks as they would probably be detected as bad users, it could be used just the contour clicks to help creating the final binary mask.

4.2. Manually categorization

Once it is defined all possible categories, it is done a manually categorization by considering just the train set, the 5 gold standard images, table 1

Users	Manually categorization
1	Painter
2	Expert
3	Mirror
4	Expert
5	Border guard
6	Expert
7	Tired
8	Border guard
9	Expert
10	Different pattern
11	Different pattern
12	Expert
13	Expert
14	Expert
15	Expert
16	Expert
17	Tired
18	Surrounder
19	Spammer
20	Expert

Table 1: Manual categorization of all 20 users

4.3. Heuristic rules for automatic user categorization

Given all commented particularities in section 4.1, it is created a set of features that will help us to distinguish two different users. Features are: number of clicks per image, percentage of foreground clicks in an image, defined as the relation between the foreground clicks and the total number of clicks in a image, error rate, jaccard index, percentage of foreground contour clicks, defined as the relation between the foreground contour clicks and the total foreground clicks and finally, the percentage of background clicks, defined as the relation between the background contour clicks and the total background clicks.

Finally, in order to ease the task of automatic user categorization, it is created the table 2 where can be seen the rules that have been set manually by looking the feature values of each user in the train set. It can be clearly seen that each user is identified with a different set of features and a different rules value. It is important to consider that in this table is not present the different pattern user, as there do not exist any feature and rule capable to describe it. Consequently, if a user does not follow any set of rules of the table will be detected as a different pattern user.

Features	Painter	The mirror	The border guard	The surrounder	The spammer	The tired	The expert
# clicks	>150/image	-	-	-	-	<5/image	-
fg clicks(%)	>95%	-	<20%	>95%	>90%	-	-
errors(%)	<3%	>90%	-	-	>40%	<20%	-
Jaccard index (%)	-	<10%	-	-	-	<80%	>80%
Contour fg(%) (fg contour clicks/total fg clicks)	-	-	-	>80%	<80%	-	-
Contour bg(%) (bg contour clicks/total bg clicks)	-	-	>70%	-	-	-	-

48

Table 2: Rules for the automatic user categorization

5. Results

Sections 3 and 4 have presented how to treat with human interactions based on the training set. This chapter presents the results in the test set taking into account all techniques introduced in previous sections. In figure 21 it is shown how will be calculated the Jaccard index reference value in the test set. In the case of presenting the results from section 3.1, for each image in the test set it is used just the information of the kept users. However, as has been explained in chapter 3.2, in case of foreground map algorithm for each image it is used all clicks from all 20 users.

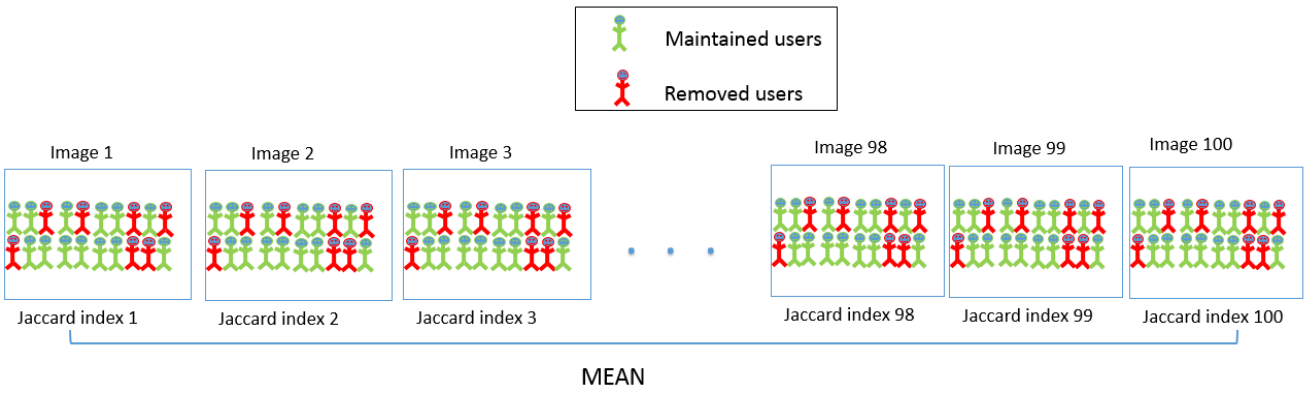

Figure 21: Jaccard index calculation for the test set

5.1. Results by filtering users

Section 3.1.1 has proposed two different approaches to separate good from bad users: error rate and Jaccard index based on the train set. Figure 22 shows the resulting curves for the test set of both approaches. It can be clearly seen that error rate is not discriminant enough as good users are not detected as the ones that have the lowest error rate. Moreover, in order to confirm this previous hypothesis, if it is focused on users between six and thirteen sorted by its error rate (green curve) it is appreciated a rise in the Jaccard index in the test set. Sorting users by its Jaccard index in the train set (blue curve) from Figure 22 shows how the best result is achieved when considering only the two best workers, with a Jaccard of 0.9 comparable to what expert users had reached (see Section 2.1.4). It could be argued that two users are not significant enough and that reaching such a high value as 0.9 could be a statistical anomaly. Nevertheless, if many more users are considered and clicks from the top half users are processed, a still high Jaccard of nearly 0.85 is achieved. Therefore, the main conclusion that can be derived from this graph is sorting users by its Jaccard index have a better performance than sorting them with their error rate.

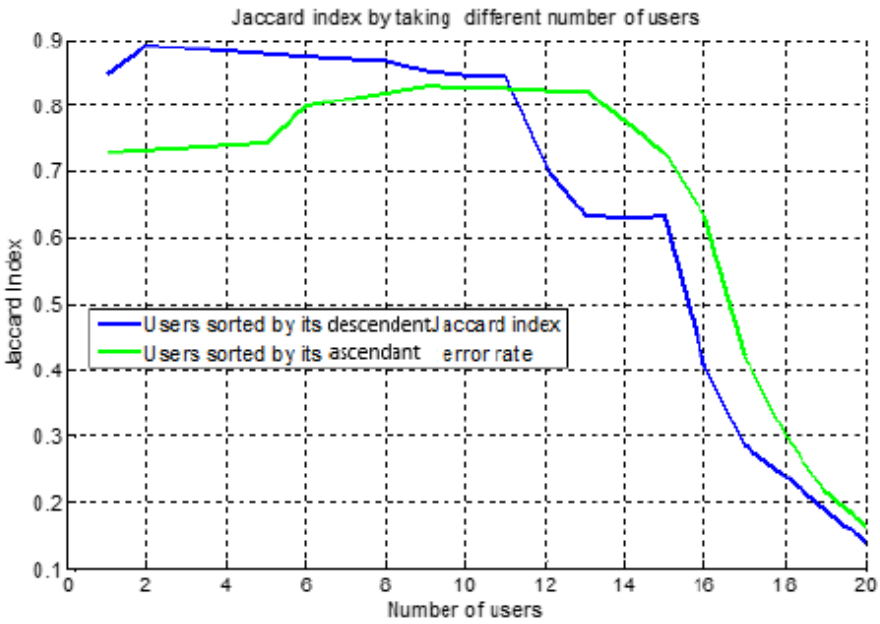

Figure 22: Results in the test set sorting users by its Jaccard index and error rate

5.2. Results by filtering clicks

Section 3.1.2 has presented two different techniques for filtering clicks in conflicting superpixels. On table 3 is shown the results both Slic and Felzenszwalb techniques. Comparing both filtering and non-filtering clicks results, it can be appreciated a slight improving by filtering clicks. However, Jaccard indexes are still too low to consider segmentations useful.

.

Without applying any technique of filtering clicks	0.14	
Techniques of filtering clicks in a same sppxl.	Partial removal of conflict clicks	Total removal of conflict clicks
SLIC	0.2109	0.2412
FELZ	0.2104	0.2240

Table 3: Results applying both filtering clicks techniques

This result indicates that filtering users has a much greater impact than just filtering clicks, as presented in Section 5.1, where the best Jaccard obtained was 0.9.

5.3. Results by filtering clicks and users

This section explores the combination of algorithm commented on sections 3.1.1 and 3.1.2 to further clean the remaining set of clicks. Regarding to the measure to remove users it will be focused just on the Jaccard index based on the train set, since in section 5.1 has been proved that it is the optimal measure to separate good from bad users. At this point, figure 23 shows the Jaccard curves obtained when applying partial filtering after user filtering. Graphs indicate that there is no major effect when considering a low number of higher quality users, but that the effect is more significant when adding worse users(approximately from user12). The case of filtering all conflicting clicks is studied in Figure 24. In this situation, this filtering causes a severe drop in performance when few users are considered, and has mostly the same effect as majority filtering otherwise. This is probably explained by the fact that discarding all clicks when few users are considered results too aggressive and does not provide enough labels to choose a good combination of object candidates.

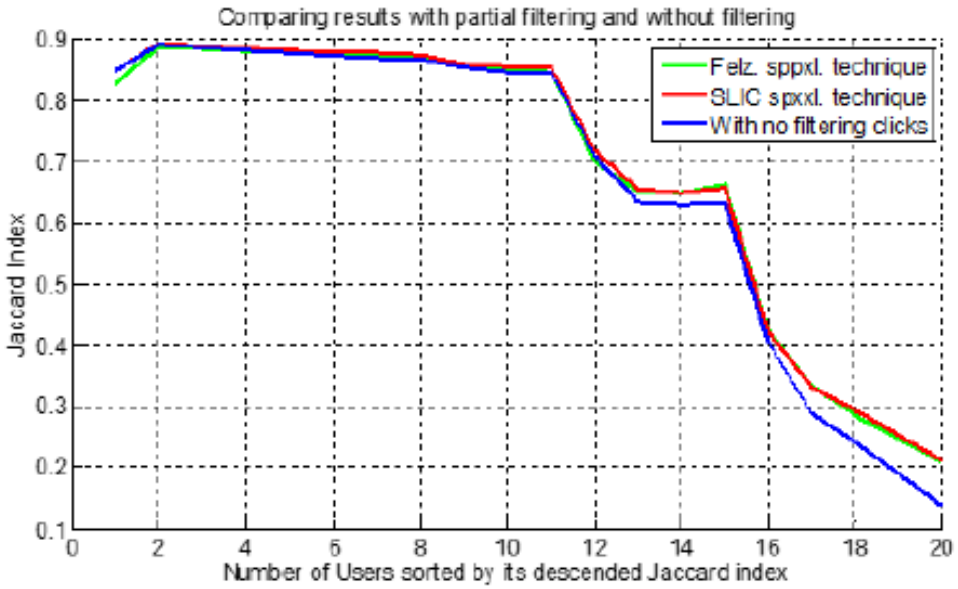

Figure 23: Segmentation results with the best N users according to their personal Jaccard-based quality estimation. Red and green curves consider filtering by majority, while blue curve does not apply any click filtering

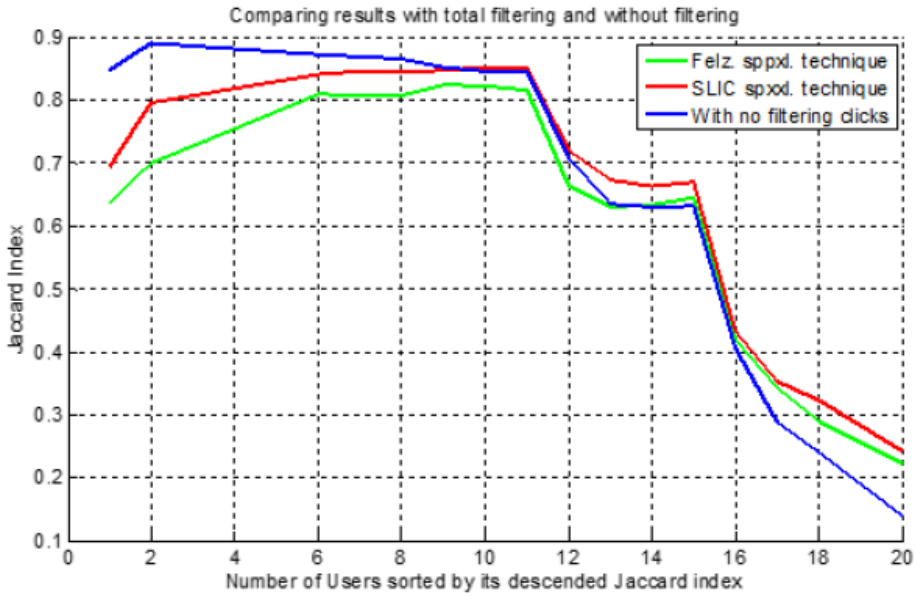

Figure 24: Segmentation results with the best N users according to their personal Jaccard-based quality estimation. Red and green curves discard all conflicting clicks, while blue curve does not apply any click filtering.

5.4. Results in the improved foreground map

In this chapter it is exploited the methodology explained in section 3.2. First of all, it is computed the foreground map in the train set (figure 25, left picture) in order to estimate the threshold that will be used in the foreground map on the test set. The value of the threshold that give us a highest Jaccard index in the test set, it will be our estimated threshold. Then, it is computed the foreground map on the test set(figure 25, right picture) and the final Jaccard index on the test set it will be given by the estimated threshold. Therefore, it is obtained a final Jaccard index value of 0.86 with the threshold equal to 0.56. It can be concluded that it is a really good value of Jaccard index taking into account that no-filtering process is present. Furthermore, if it is compared to the previous work (section 2.1.4), which the highest value reached was 0.83 when filtering worst performing users, it can be considered the foreground map algorithm as a very robust technique to face crowd users without any filtering.

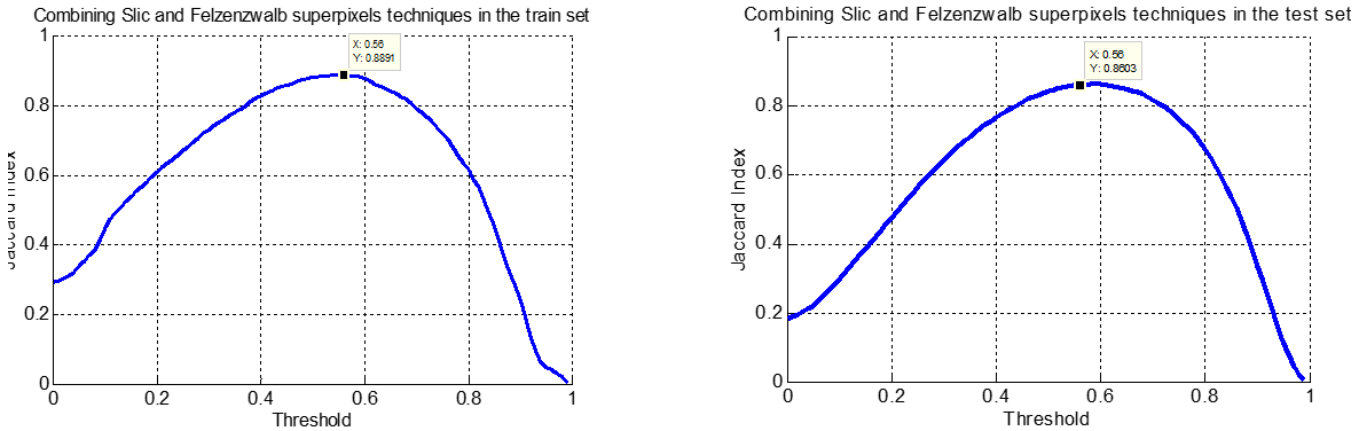

Figure 25: Foreground map combining both Slic and Felzenszwalb superpixel technqiues in the train set(left) and in the test set(right).

5.5. Evaluation of the automatic categorization of users

Section 4 has introduced table 2 with some heuristic rules that have been designed by looking at the images in the train set. In table 4, it is presented the confusion matrix for the test set in order to see how good, the rules that have been set, categorize users. It can be seen that categories like painter, mirror, spammer, surrounder and different pattern are well categorized. However, either tired, expert or border guard are not all well detected, as one user of each category is detected as a different pattern user. Despite this fact, not all bad detected categories have the same precision and recall. For this reason, table 5 contains the precision and recall for each category. It can be clearly appreciated that, in spite of one expert is detected as a different pattern user, the precision and recall of expert category keeps being high. Moreover, the recall of the border guard and tired category is not so high as well as the precision of the different pattern user.

		Prediction							
		Painter	Mirror	Expert	Spammer	Surrounder	Border Guard	Tired	Diff. Pattern
Ground Truth	Painter	1	0	0	0	0	0	0	0
	Mirror	0	1	0	0	0	0	0	0
	Expert	0	0	9	0	0	0	0	1
	Spammer	0	0	0	1	0	0	0	0
	Surrounder	0	0	0	0	1	0	0	0
	Border guard	0	0	0	0	0	1	0	1
	Tired	0	0	0	0	0	0	1	1
	Diff. pattern	0	0	0	0	0	0	0	2

Table 4: Confusion matrix of the automatic categorization rules in the test set

	Precision	Recall
Painter	1	1
Mirror	1	1
Expert	1	0.9
Spammer	1	1
Surrounder	1	1
Border guard	1	0.5
Tired	1	0,5
Diff.pattern	0.4	1

Table 5: Precision and Recall for each user category

6. Budget

In this section it is analyzed all the costs and the budget of the project. Assuming that this is a researching project, there is no final product to sell or to rent. Besides, the physical component required for this thesis is a personal computer.

The software required is Matlab and its individual license cost 2000€.

In table 6 it is shown the budget considering that the remuneration of a junior engineer is 8€ per hour.

	Remuneration per hour	Months in the project	Hours per day	Total days worked	Total hours	Total cost
Junior engineer	8€	5	7	110	770	6160€

Table 6: Project budget

The global cost of the project would be 8160€.

7. Conclusions

This work has explored different strategies to face the problem of object segmentation in crowdsourcing. Different approaches have been presented during the whole work to solve this problem. First of all, as bad users tend to create bad human interactions it was focused on detecting and separating good from bad users. Their contribution of Jaccard index in the train set has resulted to be the best measure to do this detection and separation. By taking the two users that have higher contribution of Jaccard index in the train set, it is reached a final result of 0.9, even better than the results of expert users with the same platform (0.89) [3] and comparable to results of other expert users using different tools [7] (0.93). Then, instead of removing all information of a user, it has been proposed to filter clicks based on superpixels. The proposed strategies for filtering clicks based on superpixels introduced significant gains with respect to previous work (section 2.1.4), but the final quality was still too low. Thus, as an alternative to remove bad human interaction, it was focused on improving the foreground map algorithm [24]. A final Jaccard index of 0.86 it is reached by combining different superpixel techniques, in our case, Slic and Felzenszwalb, with different parameters. It is a very significant value, even more if it is taken into account that it is not applied any filtering of neither users nor clicks. Finally, it has focused on trying to convert bad human interaction into good one. For this reason, the first step is to automatically categorize the different users using some heuristic defined rules. The confusion matrix (section 5.5) showed that expert, border guard and tired users are the only ones that give problems when automatically categorize them. The presented results indicate the potential of using image processing algorithms for quality control of noisy human interaction, also when such interaction may eventually be used to train computer vision systems. In fact, it is the combination of the crowd (majority of correct clicks) and image processing (superpixels) which allows the detection and reduction of a minority of noisy interactions.

This work has been submitted with the help of Axel Carlier, Amaia Salvador, Xavier Giró and Vincent Charvillat to IEE International Conference on Image Processing (ICIP).

8. Future development:

The result of this study has produced a slight improvement by just using two simple techniques for filtering clicks. For this reason, a further study on filtering clicks can be made to obtain even higher results.

Furthermore, as it has been exposed during the section of automatic user categorization, once a user is categorized it can be exploited their clicks to help us to create a better binary mask. In figure 26 can be depicted the clicks from border guard and surrounder users. It is a very rich information and, if it is known how to use it correctly, it can ease us obtaining the resulting binary mask.

Finally, it could be trained a classifier in order to create the rules for automatic user categorization, as they have been set manually and it is not so consistent.

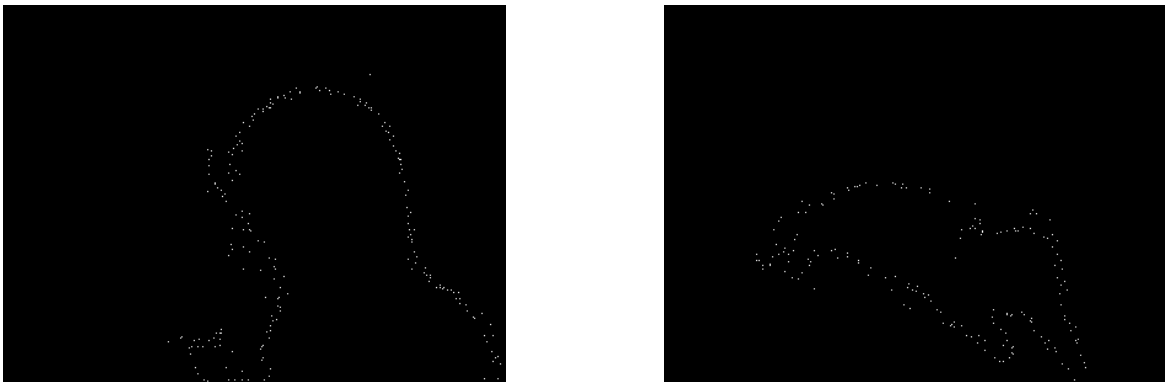

Figure 26: Contour clicks produced by surrounder and border guard users.

Bibliography:

[1] Bryan C Russell, Antonio Torralba, Kevin P Murphy, and William T Freeman, "Labelme: a database and web-based tool for image annotation," International journal of computer vision, vol. 77, no. 1-3, pp. 157–173, 2008.

[2] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Doll´ar, and C. Lawrence Zitnick, "Microsoft coco: Common objects in context," CoRR, 2014.

[3] Axel Carlier, Vincent Charvillat, Amaia Salvador, Xavier Giroi Nieto, and Oge Marques, "Click'n'cut: crowdsourced interactive segmentation with object candidates," in Proceedings of the 2014 International ACM Workshop on Crowdsourcing for Multimedia. ACM, 2014, pp. 53–56.

[4] Amaia Salvador, Axel Carlier, Xavier Giro-i Nieto, Oge Marques, and Vincent Charvillat, "Crowdsourced object segmentation with a game," in Proceedings of the 2nd ACM international workshop on Crowdsourcing for multimedia. ACM, 2013, pp. 15–20.

[5] Pablo Arbel´aez and Laurent Cohen, "Constrained image segmentation from hierarchical boundaries," in Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008, pp. 1–8.

[6] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," in ACM Transactions on Graphics (TOG). ACM,2004, vol. 23, pp. 309–314.

[7] Kevin McGuinness and Noel E. O'Connor, "A comparative evaluation of interactive segmentation algorithms," Pattern Recognition, vol. 43, no. 2, 2010.

[8] Xavier Giro-i Nieto, Neus Camps, and Ferran Marques, "Gat: a graphical annotation tool for semantic regions," Multimedia Tools and Applications, vol. 46, no. 2-3, pp. 155–174, 2010.

[9] David Oleson, Alexander Sorokin, Greg P Laughlin, Vaughn Hester, John Le, and Lukas Biewald, “Programmatic gold: Targeted and scalable quality assurance in crowdsourcing.,” Human computation, vol. 11, pp. 11, 2011.

[10] Luke Gottlieb, Jaeyoung Choi, Pascal Kelm, Thomas Sikora, and Gerald Friedland, “Pushing the limits of mechanical turk: qualifying the crowd for video geo-location,” in Proceedings of the ACM multimedia 2012 workshop on Crowdsourcing for multimedia. ACM, 2012, pp. 23–28.

[11] Hao Su, Jia Deng, and Li Fei-Fei, “Crowdsourcing annotations for visual object detection,” in Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence, 2012.

[12] Luis Von Ahn and Laura Dabbish, “Designing games with a purpose,” Communications of the ACM, vol. 51, no. 8, pp. 58–67, 2008.

[13] Andrew Mao, Ece Kamar, Yiling Chen, Eric Horvitz, Megan E Schwamb, Chris J Lintott, and Arfon M Smith, “Volunteering versus work for pay: Incentives and tradeoffs in crowdsourcing,” in First AAAI Conference on Human Computation and Crowdsourcing, 2013.

[14] Panagiotis G Ipeirotis, Foster Provost, and JingWang, “Quality management on amazon mechanical turk,” in Proceedings of the ACM SIGKDD workshop on human computation. ACM, 2010, pp. 64–67.

[15] P. Welinder and P. Perona, “Online crowdsourcing: Rating annotators and obtaining cost-effective labels,” in Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on, June 2010, pp. 25–32.

[16] Sudheendra Vijayanarasimhan and Kristen Grauman, “Largescale live active learning: Training object detectors with crawled data and crowds,” International Journal of Computer Vision, vol. 108, no. 1-2, pp. 97–114, 2014.

[17] D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring

ecological statistics," in Proc. 8th Int'l Conf. Computer Vision, July 2001, vol. 2, pp. 416–423.

[18] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," IJCV, vol. 88, no. 2, 2010.

[19] P. Arbelaez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, June 2014, pp. 328–335.

[20] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Susstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 34, no. 11, pp. 2274–2282, 2012.

[21] PedroF. Felzenszwalb and DanielP. Huttenlocher, "Efficient graph-based image segmentation," International Journal of Computer Vision, vol. 59, no. 2, pp. 167–181, 2004.

[22] J. Steggink and C. Snoek. Adding semantics to image-region annotations with the name-it-game. Multimedia Systems, 2011.

[23] L. von Ahn, R. Liu, and M. Blum. Peekaboom: a game for locating objects in images. In CHI'06, 2006.

[24] A.Carlier, Combining Content Analysis with Usage Analysis to better understand visual contents, PHD Thesis, 2014.