HCI Report One
Implementation and Design of iTag, a Image Labelling Program

Sean Wilson, Daniel Wren
s0831408, s0789266
Edinburgh University, Informatics

1. Abstract

iTag was created to be a fast and effective way for a user to be able to tag and label images. Through researching and testing of a current image labelling software called ‘LabelMe’, it was possible to gain knowledge on what features users liked and disliked in an image labelling software.

From background research it was found that users feel less overwhelmed by a new program when the graphical user interface (GUI) shares a familiar layout and familiar features to programs that they have previously used and understand.

The implemented version of iTag took into account the background research and was found to be easily understood and mastered by a novice with little practice time. A few users suggested extra features that would improve the speed of tagging an image, but also could increase complexity, and so reduce the success of iTag’s secondary goal, simplicity.

2. Introduction

2.1 Familiarity

‘Label Me’ has a familiar layout which users of many other applications will welcome. The tools and buttons for the program are located at the top of the screen, above the image panel, which is familiar to any users of commercial programs such as Microsoft programs who own a 77%[1] share of the OS market space.

2.2 Ambiguous and Hidden Features

The naming of buttons is not so familiar and intuitive. An example would be the “Erase” button. This button deletes the last point of an open polygon. The conventional name for such as function is “undo”. Erase is an ambiguous word, which is only clarified when the user hovers over the button for the tool prompt.

Features that are not clear straight away can also lead to user frustration. For example, it is not immediately clear, and not mentioned in the help dialogue, how to edit the name of an existing tag. Through a bit of exploration, the renaming feature can be found once a polygon is clicked on. Users in an unfamiliar program are unlikely to wish to explore to find the feature they are looking for, and so will assume it does not exist.

2.2 Efficient Features

‘LabelMe’ has unique features for increasing the speed in labelling and editing images by the means of:

- The “Next Image” button. This saves the current image and polygons and loads the next image. The downside to this method is being unable to select a specific directory in which to load and save the image and tag file.
- By right clicking on the image the current polygon can be closed. This is a very hand short.
- Hovering the mouse over the tag name in the list will highlight the corresponding polygon, and vice versa. This is a handy use of changing visualisation to attract the users attention, to help identify the correct object.

2.3 Degrees of Freedom

The user is given a small degree of freedom to edit the tags once they are created:

- The user can edit the points in a polygon by clicking on each point and dragging them in different direction.

- The user is also able to change the name of a closed polygon by clicking on the polygon's name or area. As mentioned previously, this is not mentioned in the help file and can only be found through using intuition and observation.

On the opposite side of the spectrum, there are a few things that limit the user for choice, which can be frustrating:

- The user must name each individual tag once a tag is completed. A default tag could be assigned if the user did not wish to name the tag at that time.
- The user cannot select the colour of the polygon being drawn, at either the beginning or while editing the polygon.
- The user is unable to add points or delete points within a polygon once closed.
- Users are unable to delete tags made by others. This is a good safe guard against malicious intent, but can be frustrating if an image is incorrectly labelled and the user is unable to change the incorrect tag.
- The user has no option, before or after tagging, what the colour of the tag will be. This is changed automatically for each new user.

2.3 A Clear and Clean Image Labeller

From the background research into the features of LabelMe, and by using the principals of Human Computer Interaction, iTag will be designed with the user in mind. The primary ambition that is to be achieved through iTag is to produce a fast and efficient method of labelling images. The secondary goal for iTag is to be an image labeller in which a novice user is able to tag images without any previous help or introduction through simplicity and familiar features to any user of other programs.

2.4 HCI Principles

The 8 Golden Rules of Shneiderman (1987) are a good list of principles to follow in the design of the iTag GUI. Shneiderman's rules also have a large cross over with Norman's 7 Principles (1988) and Nielsen's 10 Usability Heuristics. All three will be referred to throughout the explanation for iTags implementation. (For a full list of the principles, see Appendix A).

3. Method

3.1 *GUI layout and Naming*

3.1.1. Familiarity

The concept of declarative memory models, particularly schemas were taken into consideration while designing the iTag interface. The layout of iTag was designed to be familiar to any computer user. The layout of a program, if familiar can trigger scripts in memory when put in a similar scenario encountered in other programs. For example, have a tool bar running along the top with drop down menus labelled 'File' and 'Help' were carefully chosen to have a familiar naming scheme and order to current programs. If the user is asked to save and image or open a new image, they will most likely click the file drop down menu.

3.1.2. Clear and Simple

The GUI and features have been left as minimalist as possible (see Appendix B), due to the principals of perception, users can only notice detail in a small focusable area, and so have many different complex icons in a GUI could turn away a user. As shape recognition for a human has the second largest field of view, it was decided that grouping similar shaped objects together would help the user identify what they need. As such all buttons are located together, as are lists.

Image tagging software does not need to be overly complex as there are only a few basic features needed to tag fast and effectively. Each of the tools used for the tagging of an image are located on the right hand side, with clear and precise names. All tools regarding image loading, saving and closing are located under the familiar drop down menu, 'file'.

3.2 *Helping the User*

3.2.1. Welcome Image

Through a background survey carried out with 50 people, it was found that only 5% of people read a manual for a new piece of software and

30% would locate the help file within a program when needed. 96% would prefer an introduction when opening a program for the first time in the form of a step by step tutorial.

A simple images is opened in the image panel when the program is started (see Appendix C). This is a clear informal image with three simple steps to loading and tagging an image. This image will be loaded again if a user closes their image.

3.2.2. Help Box

Located in the tool bar is a help drop down menu which give the user the option of 'help' and 'about'.

'Help' opens a new frame with complete in-depth instructions on how to perform every feature in iTag. 'About' loads up contact information and version number for the software.

3.2.3. Prompting the User

Norman's 6th principal, "Design to Fail", and Shniederman's 4th rule were taken into account with the design of user prompting. To ensure no accidental lose of work, dialogue boxes, asking the user if they wish to save when the user tries to either open a new image, close an image or exit the program (Appendix D). If no changes to an image are done, or no image is loaded the user is only asked to confirm that they wish to exit the program. The user is also prompted before deleting a tag. To ensure that the user is deleting the correct tag, the name of the tag that will be deleted is displayed for the user to confirm.

If the user hovers the mouse icon above a button, a tool prompt will pop-up giving further information on the button's use.

3.3 Loading and Saving

3.3.1. Location and Serialisation

The main disadvantage to LabelMe is the inability to load a specific image. This feature has been added to iTag. When a user clicks the open button, Java FileChooser opens and allows the user to browse for a specific image. The image panel has a size of 640x480 and any image larger than this will be resized to fit this panel. If an image is smaller, it will simply be centred to avoid any image aliasing.

When a user clicks save, the created tag list is serialised and saved in the same directory as the image, with the name:

```
imageFileName – iTags.ser
```

The user is not give the option to edit the serialised file name so as to simplify the saving process for the user.

One serialised tag list can be saved per image. Upon reopening the image the corresponding .ser file (if one exists) will be loaded, painting the tags on the image panel, the tag names in the tag list and displaying the saved colour for each individual tag.

3.4 Tagging an Image

3.4.1. Drawing, Colour and Naming

Drawing a tag on an image will be done by simply clicking on the image to create a point. Each point drawn will join to the previous point by a simple thin line. A thin line was chosen to stop any possibility of the tag lines obscuring the image. The colour of the tag can be chosen via the colour select drop-down list.

A tag is completed by clicking the 'finish tag' button. This connects the last and first point together to complete the polygon. When a user finishes a tag, the user is prompted to give the tag a name. If the user cancels or does not enter a name, the tag is given a default name of 'tag *index*', for example 'tag 2' for the second tag created. This was implemented to speed up tagging, if a user is not interested in naming the tags.

Each point added is clearly visible as a small dot. This was implemented to easy the implementation of future features, such as moving individual points.

3.4.2. The Tag List

The user of an image labelling software need a method in which to quickly identify and edit the tags on an image. In iTag this is achieved through the 'Tag List'. Each tag's name, in order of which it was created, is added to the list. When a tag name is clicked, the corresponding polygon's lines are thickened, to increase notification(Appendix F). It was considered to fill in the polygon with an opaque colour, as seen in LabelMe (Appendix E). But from the opinion of the users in our survey believe that this obstructs the actually tagged part of the image. LabelMe also has bold lines for all tags which, while tagging small parts of an image can also obscure the image. This was taken into account in iTag, and so only highlighted tags have bold lines (Appendix F).

3.4.3. The Undo Feature

From the survey results of the 50 users who tried LabelMe, over half struggled to identify the meaning of the erase button and believed it to erase the polygon itself, and not the previous point.

As with standard naming conventions found in most programs that a user is likely to have used, the button includes the work 'undo' and the topic of which is to be undone. This will have clarify, and should hopefully not leave a user uncertain.

3.4.4. Deleting a Tag

As mentioned before, unless the user reads the help file, it is unclear how to delete a tag without exploring the program. It was clear to ensure a user could use iTag without resorting to the help file, that all basic features should be added to the GUI as clearly labelled buttons. This will eliminate the need to explore, following

3.4.5. Editing a Tag

A user is able to rename or change the colour of a tag quickly and clearly by use of the 'rename' button and colour list. By clicking on a tag name in the tag list the current colour of the tag will be selected in the colour list. By changing the colour in the list, the tag will change colour respectively.

4. Results

Question	**LabelMe** (yes/no/unsure)			**itag** (yes/no/unsure)		
	Y	**N**	**U**	**Y**	**N**	**U**
Draw new tag	76%	22%	2%	74%	24%	2%
Delete tag	12%	74%	14%	92%	8%	0%
Remove point	52%	32%	26%	96%	4%	0%
Open image	94%	6%	0%	98%	2%	0%
Highlight tag	86%	8%	6%	70%	30%	0%

Figure 1: LabelMe and Itag GUI image questionaire on guicheck.com

Question	**iTag**		
	Yes	**No**	**Unsure**
Draw new tag	98%	2%	0%
Delete tag	100%	0%	0%
Remove point	100%	0%	0%
Open image	100%	0%	0%
Highlight tag	88%	12%	0%

Figure 2: iTag second run after users have seen the welcom screen

The results for the guicheck.com results can be found at:

iTag results:
http://guicheck.com/surveys/6e5113/results.html#
LabelMe results:
http://guicheck.com/surveys/6bb143/results.html#

From the results gained from the 50 test users (shown in figure 1) who contributed on-line, users were also less inclined to click on the 'unsure what to do' button with iTag. With LabelMe, users were more inclined to click the unknown action option with question which required exploration of the GUI, for example removing a tag which requires the user to first select a tag (an example of the outcome of the guicheck can be seen in Appendix G. This is the results for question 5).

The lowest result for iTag was for highlighting a drawn tag. 30% of the users clicked on the polygon itself, which thus far is not an implemented feature (Appendix G). The lowest result for LabelMe was deleting a tag. 14% were unsure of where to click and 74% of people incorrectly clicked the ‘erase’ button.

After the 50 users were shown the welcome screen (see Appendix C) no user clicked the unsure option for any of the question. Users still were sometimes inclined to click on the polygon to select a current tag.

5. Discussion

The survey for iTag at guicheck.com was taken by users who did not have the option to see the welcome screen or open the help function. Fewer users of iTag were inclined to click the unknown option box compared to LabelMe. The reason behind this is because of iTag’s principle, taken from Norman’s 2nd principle, “Make things visible”. If a user does not need to explore, they are more likely to try the available options, especially if the option seems viable.

Once the users were opened to the welcome screen they seemed to respond well to the short and visually pleasing entry screen and so felt enticed to read it. This helped all user to correctly answer the original five questions almost 100% correctly.

It was clear from the results of question 4 that using a familiar layout works as, although no user of the guicheck questionnaire had access to the program or help function, 98% of the users clicked the file drop down menu to open a new image. This shows that when not explicitly told what to do, user intuition and previous experience takes over, especially in a familiar layout.

iTag has an increase in usability over LabelMe it terms of user previous knowledge. iTag is based on the principle of users’ knowledge of existing software, where as LabelMe requires the user to either read the help text or explore further into the GUI, which can be daunting for some users.

From user feedback it is clear that the most confusing feature is selecting previous tags. Many users felt it would be a handy feature to be able to select a previous tag by clicking on the actual polygon on the image. This feature was considered, but left out for good reason. LabelMe uses this feature and can be frustrating if a new tag will start on the edge of a current tag. Instead of starting a new tag, the existing tag is highlighted for editing. A further feature that could over come this issue is having two radio buttons that toggle between ‘Edit Mode’ and ‘Tag Mode’. This will allow the user to choose between the two without adding complexity to the program.

Another feature implemented in LabelMe that will be added in a future version of iTag is the ability to edit current labels via moving points. This idea can be extended by adding the ability to add new points between existing points. This will increase user flexibility, but also increase the complexity of iTag.

A future feature, and Shneiderman’s 2nd rule, would be to implement short-cut features for frequent users. This would further speed up the time in which one can tag an image.

6. Conclusion

Users of iTag found that having all tagging tools visible in the GUI proved helpful. The familiar layout also allowed user intuition to take over to save, load and close an image.

Giving the user more features such as further tag editing features could be implemented, taking into account the problems discussed in the LabelMe’s implementation.

7. References

Wikapida (2011), OS Market Share, http://en.wikipedia.org/wiki/Usage_share_of_operating_systems

8. Appendix

Appendix A – Design Rules for HCI

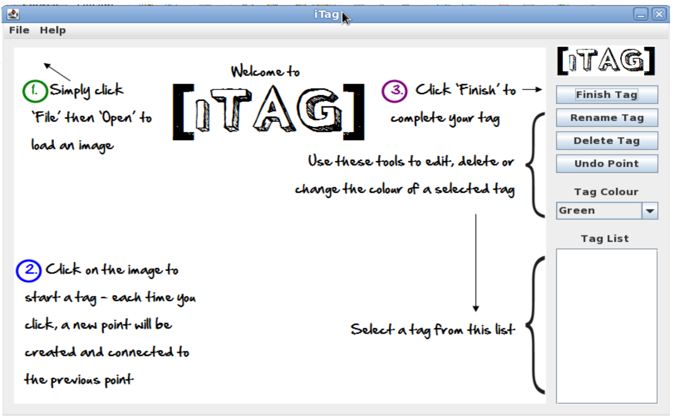
Shneiderman’s 8 Golden Rules (1987):
1. Strive for consistency
2. Enable frequent users to use shortcuts
3. Offer informative feedback
4. Design dialogs to yield closure
5. Offer error prevention and simple error handling
6. Permit easy reversal of actions
7. Support internal locus of control
8. Reduce short-term memory load

Norman’s 7 Principles (1988):
1. Use both knowledge in the world and knowledge in the head.
2. Simplify the structure of tasks.
3. Make things visible.
4. Get the mappings right.
5. Exploit the power of constraints, both natural and artificial.
6. Design for error.
7. When all else fails, standardize.

Nielsen’s 10 Usability Heuristics (1994):
1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Help users recognize, diagnose and recover from errors
6. Error prevention
7. Recognition rather than recall
8. Flexibility and efficiency of use
9. Aesthetic and minimalist design
10. Help and documentation

Appendix B – The iTag GUI

The GUI of iTag shown as a user would see when first opening the program. As can be seen the layout is simple and based on well known current user interfaces.



1. Simply click 'File' then 'Open' to load an image

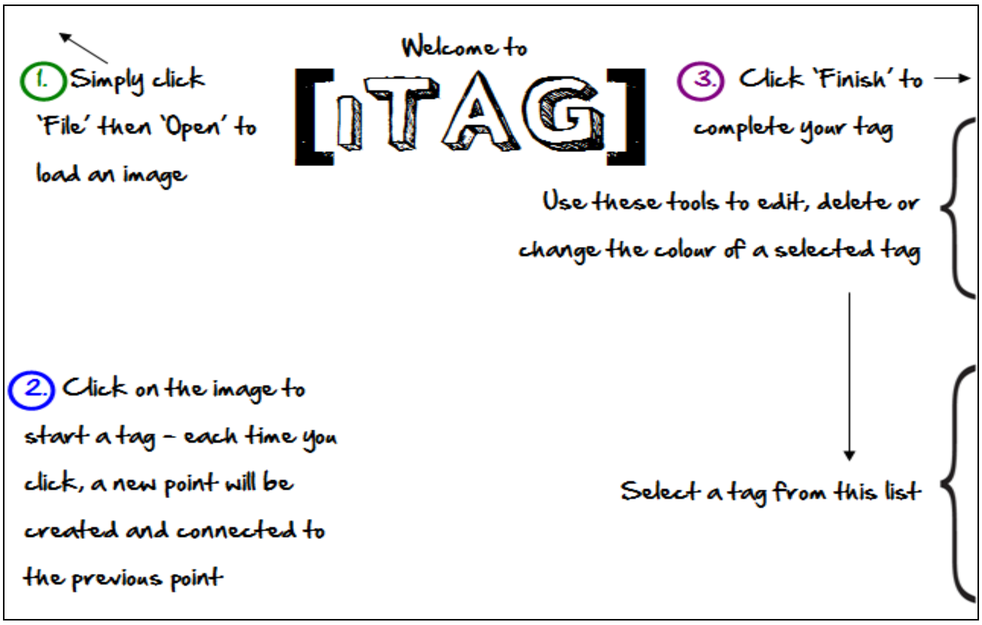
2. Click on the image to start a tag – each time you click, a new point will be created and connected to the previous point

3. Click 'Finish' to complete your tag

Use these tools to edit, delete or change the colour of a selected tag

Select a tag from this list

Appendix C – Welcome and Introduction Page



Welcome to [iTAG]

1. Simply click 'File' then 'Open' to load an image

2. Click on the image to start a tag – each time you click, a new point will be created and connected to the previous point

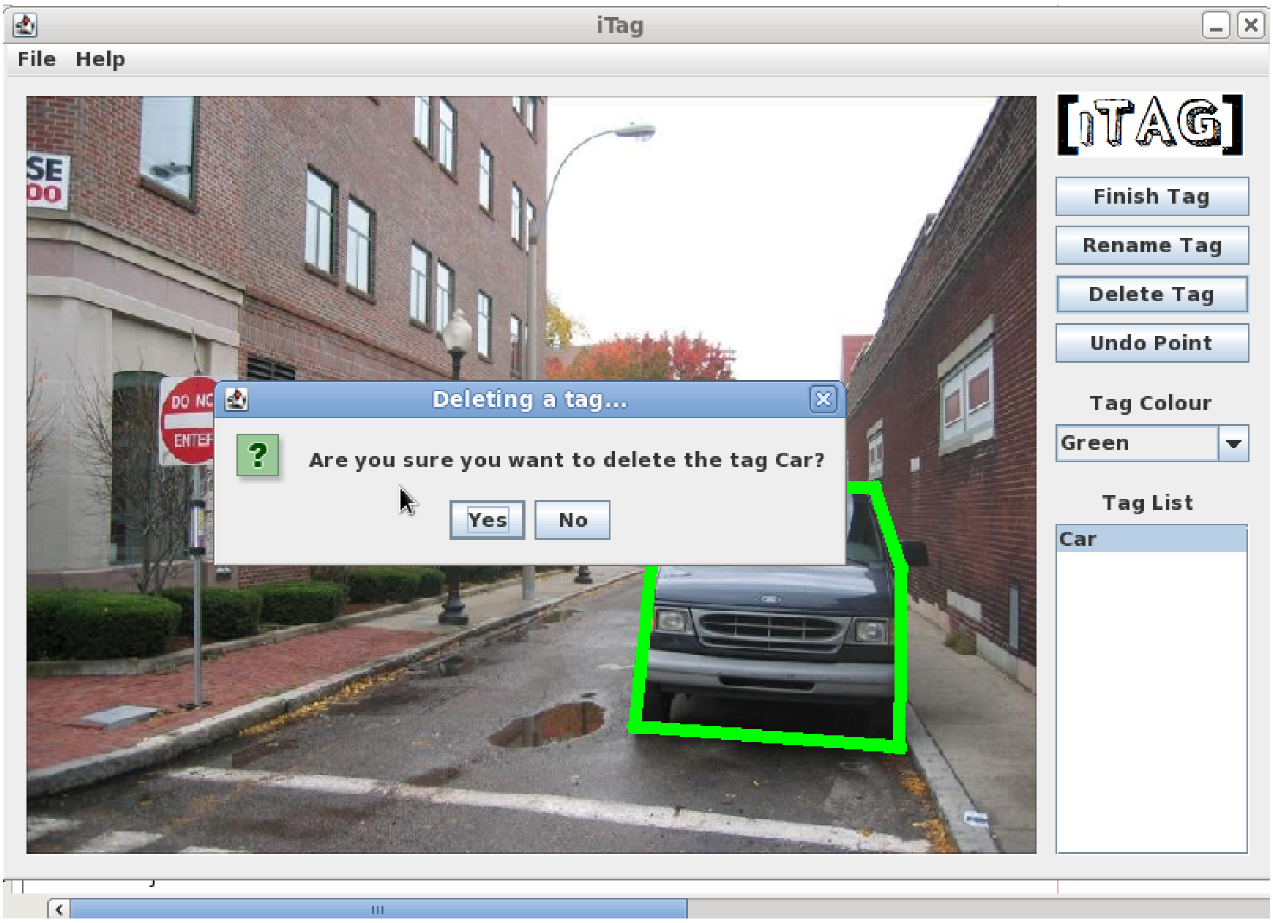
3. Click 'Finish' to complete your tag

Use these tools to edit, delete or change the colour of a selected tag

Select a tag from this list

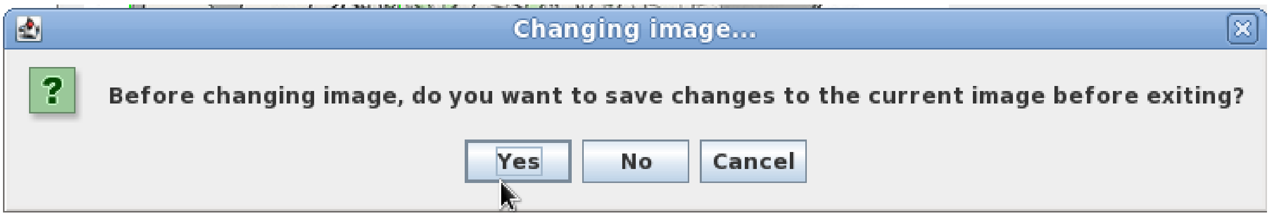
This image clearly states the three steps to start tagging an image. This is used as a quick introduction to help users familiarise with the program.

Appendix D – User prompt dialogue box



Are you sure you want to delete the tag Car?

This shows how the user is prompted when trying to delete the selected tag.

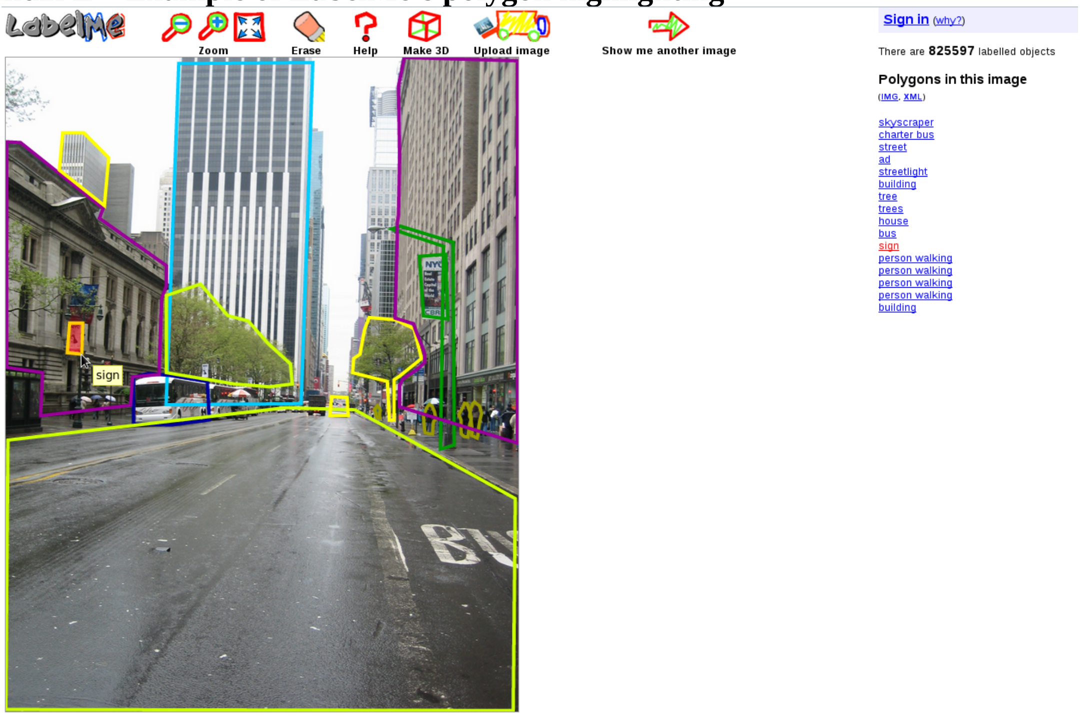


Changing image...

Before changing image, do you want to save changes to the current image before exiting?

This is the prompt the user will see when trying to open up a new image to ensure they have saved their work.

Appendix E – Example of LabelMe's polygon highlighting



Sign in (why?)

There are 825597 labelled objects

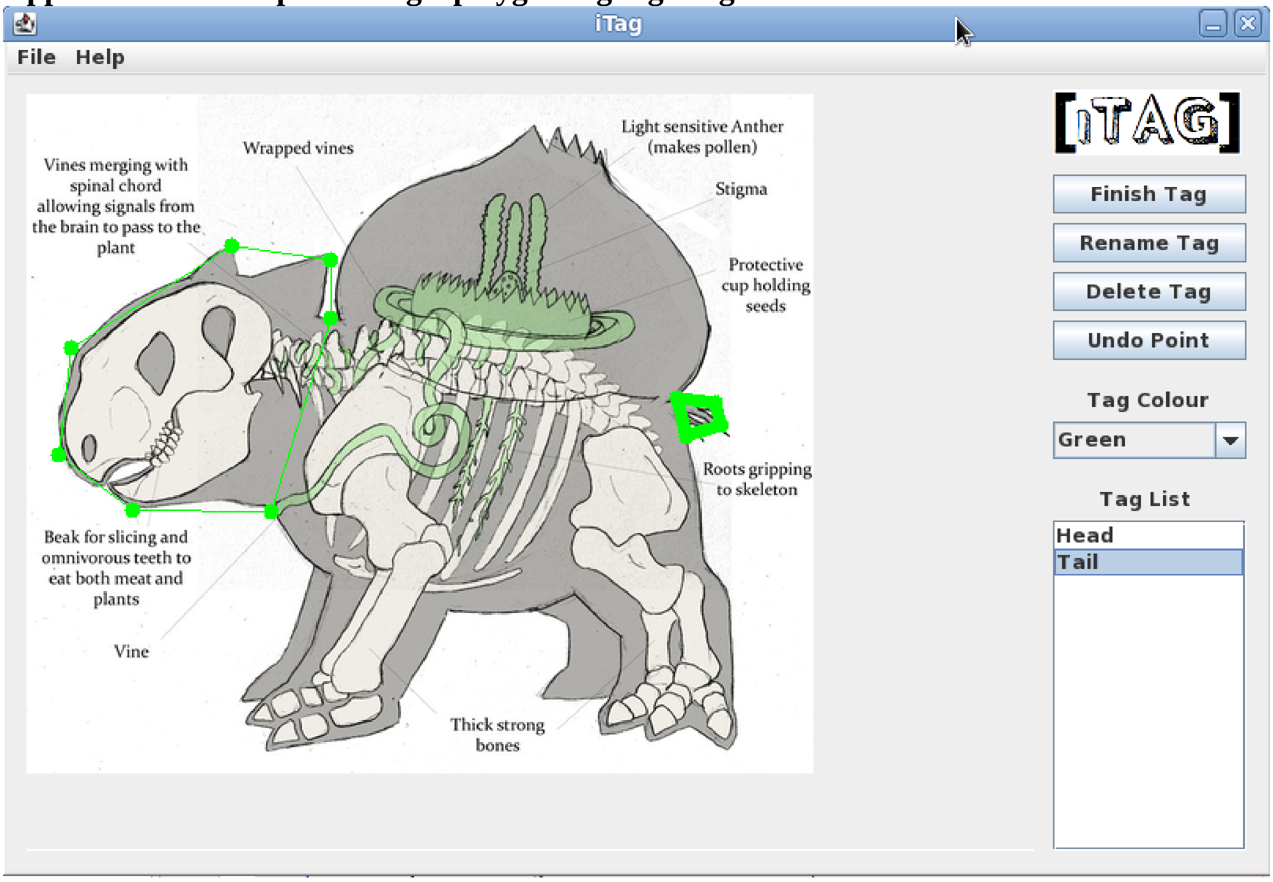
Polygons in this image

(IMG, XML)

- skyscraper
- charter bus
- street
- ad
- streetlight
- building
- tree
- trees
- house
- bus
- sign
- person walking
- person walking
- person walking
- person walking
- building

As can be seen in this image, the people right of the screen, the bold lines make it difficult to identify the tagged image as people. The selected polygon, 'sign' is highlighted in red, which makes it difficult for the user to see what the actual tagged part is.

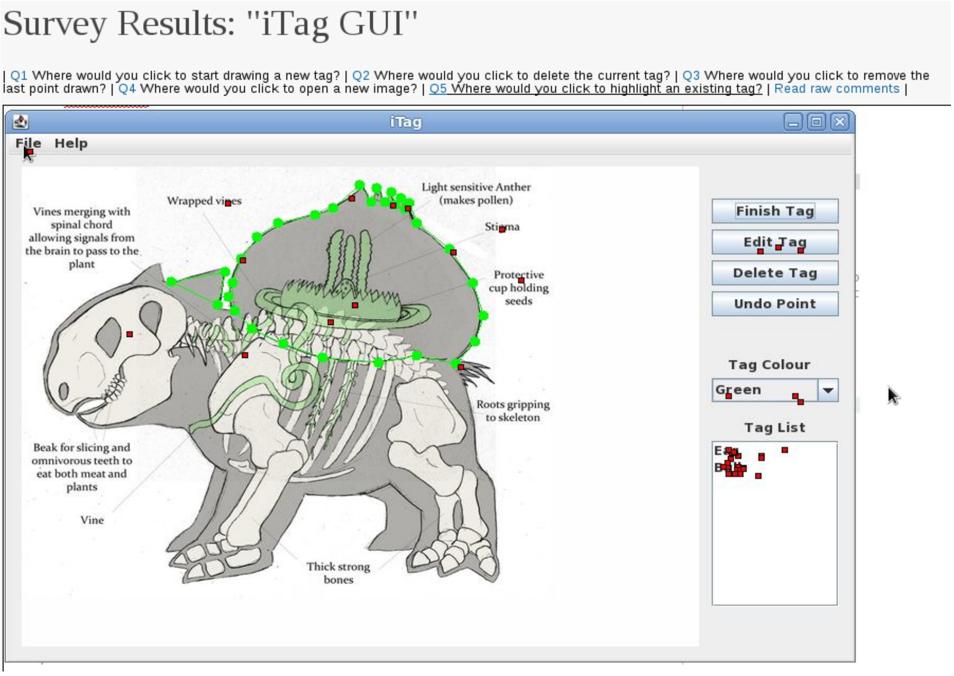
Appendix F – Example of iTag’s polygon highlighting


iTag
File
Help
Vines merging with spinal chord allowing signals from the brain to pass to the plant
Wrapped vines
Light sensitive Anther (makes pollen)
Stigma
Protective cup holding seeds
Roots gripping to skeleton
Beak for slicing and omnivorous teeth to eat both meat and plants
Vine
Thick strong bones
[iTAG]
Finish Tag
Rename Tag
Delete Tag
Undo Point
Tag Colour
Green
Tag List
Head
Tail

As can be seen, the tail tag is selected in the tag list and the corresponding tag has been turned bold in response.

Appendix G

As can be seen in this example, guicheck surveys show where users click in response to the question. As can be seen here the majority of users click in the correct location, the tag list. But 30% of the users clicked the actually tag on the image.



Survey Results: "iTag GUI"

| Q1 Where would you click to start drawing a new tag? | Q2 Where would you click to delete the current tag? | Q3 Where would you click to remove the last point drawn? | Q4 Where would you click to open a new image? | Q5 Where would you click to highlight an existing tag? | Read raw comments |

iTag
File
Help
Vines merging with spinal chord allowing signals from the brain to pass to the plant
Wrapped vines
Light sensitive Anther (makes pollen)
Stigma
Protective cup holding seeds
Roots gripping to skeleton
Beak for slicing and omnivorous teeth to eat both meat and plants
Vine
Thick strong bones
Finish Tag
Edit Tag
Delete Tag
Undo Point
Tag Colour
Green
Tag List