Smart cropping tools with help of machine learning

John Kanwar

Computer Game Programming, bachelor's level
2019

Luleå University of Technology
Department of Computer Science, Electrical and Space Engineering
Abstract

Machine learning has been around for a long time, the applications range from a big variety of different subjects, everything from self driving cars to data mining.

When a person takes a picture with its mobile phone it easily happens that the photo is a little bit crooked. It does also happen that people takes spontaneous photos with help of their phones, which can result in something irrelevant ending up in the corner of the image.

This thesis combines machine learning with photo editing tools. It will explore the possibilities how machine learning can be used to automatically crop images in an aesthetically pleasing way and how machine learning can be used to create a portrait cropping tool. It will also go through how a straighten out function can be implemented with help of machine learning. At last, it is going to compare this tools with other software automatic cropping tools.

Sammanfattning

Maskinlärning har funnits en lång tid. Deras jobb varierar från flera olika ämnen. Allting från självkörande bilar till data mining.

När en person tar en bild med en mobiltelefon händer det lätt att bilden är lite sned. Det händer också att en tar spontana bilder med sin mobil, vilket kan leda till att det kommer med något i kanten av bilden som inte bör vara där.

Acknowledgements

I would like to thank Once Upon who gave me the opportunity to make my thesis at their company. I would also like to thank my supervisor Patrik Holmlund at LTU who helped me with the structure of the report and its writing. Thanks to Lina Näsström who gave valuable tips on what makes certain pictures look more appealing than others. And thanks to Jonathan Öhman who was great to bounce ideas with.
# Contents

1 Introduction .................................................. 1
   1.1 Purpose and goals ........................................ 1
   1.2 Background .............................................. 1
      1.2.1 Machine learning .................................. 1
      1.2.2 Android ........................................... 2
      1.2.3 Android Studio .................................... 2
      1.2.4 Firebase Machine learning-kit ................. 2
      1.2.5 React Native ...................................... 2
   1.3 Related work ............................................ 3
      1.3.1 Imaggas API ....................................... 3
      1.3.2 Adobe Lightroom .................................. 3
      1.3.3 iPhone photo app ................................ 3
   1.4 Social, Ethical, and Environmental Considerations .... 3

2 Design and implementation .................................... 4
   2.1 Structure ................................................ 4
   2.2 Implementation ........................................... 4
      2.2.1 Aesthetically pleasing .......................... 4
      2.2.2 Face recognition ................................ 5
      2.2.3 Portrait cropping ................................ 5
      2.2.4 Straight out function .......................... 5
      2.2.5 Java module ....................................... 6

3 Result ........................................................ 7
   3.1 Portrait cropping ........................................ 7
      3.1.1 My portrait cropping ........................... 7
   3.2 Straighten out function ................................ 9
      3.2.1 My straighten out function .................. 9

4 Analyses and Discussion ........................................ 12
   4.1 Result .................................................... 12
   4.2 Challenges ............................................... 12
      4.2.1 Subjective thinking ............................ 12
      4.2.2 Layout ............................................ 12
   4.3 Restrictions .............................................. 12
      4.3.1 Portrait cropping ................................ 12
      4.3.2 Straight out function ........................ 12
   4.4 Related work ............................................. 13
      4.4.1 Imaggas application programming interface ... 13
      4.4.2 Comparing portrait croppings ................ 14
      4.4.3 Adobe Lightroom ................................ 15
      4.4.4 iPhones straighten out application ........... 16
      4.4.5 Comparing straight out functions .............. 17
   4.5 Future .................................................... 17
      4.5.1 Portrait cropping ................................ 17
1 Introduction

Artificial intelligence (AI) is a software that makes its own decisions.[32]. It has existed for a long time and knowledge about AI has grown with the years[15]. Machine Learning is an application of artificial intelligence that learns through analyzing data and can in that way provide solutions without being explicitly programmed[6].

In 1952 Arthur Samuel[34] began working on some of the very first machine learning programs that played checkers[10]. In 1959 he coined the term “Machine learning”[25]. Today we have machine learning in everything from self driving cars to face recognition[17][7]. Machine learning is getting more and more into our everyday life[26].

Once upon [33] is a company that lets users create photograph albums through their application. Through simplicity and ease, users can use different layouts and craft their own photograph albums. The goal of this thesis is to explore how machine learning could be used to solve manual image cropping, and instead crop images automatically but still preserve the aesthetically pleasing in the images. The other goal of this thesis is to create and implement a straighten out function.

1.1 Purpose and goals

Once Upon’s application focuses on simplicity, but with simplicity come regulations. One of those regulations is cropping images. At present, the cropping process will capture images to a size that fits a layout. This means that their current cropping process will not take people’s faces into account or objects in that matter. This can result in images having parts of faces or objects missing. Another problem is that mobile photographers tend to take photographs that are tilted. A solution to that problem is a straighten out function.

The goal of this paper is to:

- Crop images with the help of face recognition.
- Crop images in a way so it is aesthetically pleasing.
- Straighten out images that are tilted with help of face recognition.

1.2 Background

The background section provides information about different tools, platforms and methods that were used during the projects time.

1.2.1 Machine learning

Machine learning[6] (ML) is an application of artificial intelligence which job is to solve problems with ”learn from experience” as mindset. It is a generic system that isn’t programmed particularly to solve a problem, but instead improves and learns from experience to solve a problem.[6] ML algorithms don’t rely on predetermined calculations as a structure, instead they ”learn” information by using computational methods[21]. As the number of learning samples increases, the algorithms enhance their performances.

Machine learning has become bigger and bigger by the years[13]. Nowadays it is a key feature to solve many problems[13]. In this thesis, machine learning is the key feature to implement face recognition.
Deep learning

One machine learning technique is called Deep learning. Deep learning is a system that learns by example. It is a popular technique to use to teach a system to recognize and separate different objects. For example, face recognition is commonly based on deep learning. It is also used in Tesla’s driverless cars, making it able for them to recognize signs and to spot the difference between a pedestrian from a lamppost.

Deep learning models usually learn by going through a big pile of labeled data and adjusting its neural network little by little. This results in it getting closer to distinguish what the object represents.

When a deep learning model is trained to recognize a human face, it will be trained to automatically identify different facial features of a face. If you would to put similar images of a person through a deep learning model it would generate a similar output, but if a different image would be used, the output would be vastly different.

1.2.2 Android

Android is a mobile operating system developed by Google. The operating system is based on a modified version of the Linux-kernel. It was originally developed by Android Inc, but was in 2005 bought by Google. Nowadays it is estimated that 75% of the mobile systems around the world are powered by Android software. 88 percent of all smartphones that were sold in the second quarter of 2018 were powered by android software. Once Upon’s application is available on Android.

1.2.3 Android Studio

Android Studio is the official integrated development environment (IDE) for Google’s Android operating system. It is built on JetBrains’ IntelliJ IDEA software and designed specifically for Android development. Android Studio has been used in this project to build the application.

1.2.4 Firebase Machine learning-kit

Firebase Machine learning-kit (ML-kit) is a mobile software development kit. It gives Android and iOS-developers access to machine learning tools. Firebase ML-kit was launched in May 2018 and is still in beta. The face recognition that is used in this project is implemented with help of Firebase ML-kit.

1.2.5 React Native

React Native is a framework that let developers develop natively rendering mobile applications to Android and iOS. It is cross platform for both Android and iOS. It started as a hackathon project by Facebook and is now one of the most popular frameworks by big applications. A hackathon project is an event that last for several days, in which a large group of people get together to solve problems. The code in this project was written in React Native.
1.3 Related work

All this related work results are going to be compared later on in the "Analyses and Discussion" chapter. This includes Imagga’s portrait cropping, Adobe Lightroom’s straighten out function and iPhone’s photo app straighten out function. They are also going to be discussed further down in "Analyses and Discussion".

1.3.1 Imagga’s API

"Imagga is a Platform-as-a-Service providing an image recognition API that helps businesses understand and monetize image content in the cloud and on-premise"[11]. It is an application programming interface that let developers get access to face recognition among other things. They developed a portrait cropping tool that is used as a reference and is going to be compared to this thesis portrait cropping tool.

1.3.2 Adobe Lightroom

Adobe Lightroom[18] is a powerful photo editing program created by Adobe. It has a log of photo editing tools that let the users enhance images. One particular function Lightroom has is the "auto straighten image" tool. It is going to get further reviewed later on.

1.3.3 iPhone photo app

iPhone[2] has a lite pre-installed photo editing program[4]. It detects if an image is crooked and automatically straightens it out. If it is not enough, the user can manually adjust the last bit of the rotation to get the desired result. That function is going to get compared with this thesis’s auto straighten out function.

1.4 Social, Ethical, and Environmental Considerations

This project is implemented on a individual branch on Once Upon’s Git repository. This is to avoid conflicts with the main code. This project does only use royalty free images to test different cropping techniques on. Which means no users privacy got compromised while testing different cropping techniques.

Digital programs don’t affect the environment directly. However, the hardware that is used in mobile phones does have a big impact on the environment. This particular application is not performance heavy, so the users don’t need to invest in a new mobile phone to run this application.
2 Design and implementation

2.1 Structure

The program is written in React Native and consists mainly of Javascript code. Sometimes the application needs access to a platform API that React Native doesn’t have a corresponding module to. In order to do that, Java modules[23] are used. Java Modules are used to access platform application programming interfaces that React Native doesn’t have access to. Firebase machine learning kit is a module that is written in Java and implemented as a Java module. In fact, every machine learning aspect is written in Java and the cropping parts are written in Javascript.

2.2 Implementation

The following tools have been implemented in a separate version of Once Upon’s application. The images that the tools have been tested on are images with royalty free license. The tools are currently implemented within a debug standpoint. They don’t have separate buttons to be toggled on or off. Right now portrait cropping or straighten out function is always activated in the application.

2.2.1 Aesthetically pleasing

The basic cropping techniques does not take into account if an image is aesthetically pleasing. There is some ground guidelines that make an image stand out from the rest. For example is the "rule of the thirds"[24] a good guideline to use. The ”rule of thumb” that is called rule of thirds is a grid with nine equal boxes that are divided by two equally spaced horizontal lines and two equally spaced vertical lines. This can been seen in figure 1. This grid can help an image to be enhanced. The intersections of the grid is where the viewer tend to lay focus. Mainly in portraits, the eyes should be in the intersections, so the viewers focus is naturally target on the faces.

![Figure 1: Rule of thirds](image-url)
2.2.2 Face recognition

Face recognition is implemented with help of the Firebase Machine learning kit application programming interface. Face recognition is the ability to identify one or more faces in an image. It detects faces in an image by comparing and analyzing different facial patterns. It can then extract information about the faces. Part of the information includes coordinates of the face related to the image and rotation of the face. Face recognition is the base pillar of the other cropping techniques.

2.2.3 Portrait cropping

The cropping part is based on the information it collects from the face recognition. It will take the coordinates from every face, calculate the min and max of every face and then create a common bounding box around all the faces. The bounding box will have a margin from the faces, so it doesn’t crop out some part of the face. It also adjusts the motives so that they end up in the intersections of the rule of thirds. This can be seen in figure 2. The red boxes in figure 2 are what the facial recognition detected. The green lines are the margins that are created from the bounding box. The purples line is the final result.

![Figure 2: Bounding box.](image)

2.2.4 Straight out function

The straight out function is based on the rotation of the faces in the images. If there are many different directions of the faces in the image, it will not straight out the image because it would risk one or many faces in the image to look disoriented. This can be seen in figure 3. If there are many faces in the image that have the same direction of rotation, it will calculate the sum of those faces rotation and rotate the image in the opposite direction. This means that the faces will be aligned in a straight line. It will then estimate to retain as much image as possible, which means it will try to crop as little as possible, as shown in figure 4. By calculating the intersection points between the rotated image and the diagonal lines and then take the shortest distance from the intersection.
points to the middle of the image, it is possible to determine which intersection point to use to get the maximum size of the image and still keep the same ratio. This can been seen in figure 5.

2.2.5 Java module

Firebase machine learning kit is written in Java, and is not a corresponding module React Native has access to. In order to get access to this functionality, a Java module must be implemented. In order to implement a Java module, three different files have to be created. One file with all the functionality like face recognition function, another file on the javascript side that receives the information from the Java module, and the last file is the bridge that makes it possible for the Java module to send information and communicate to the React Native Javascript.
3 Result

This section includes the result after various images have been cropped by portrait cropping and straighten out. It includes the results of this thesis tools.

The result of the project was two different cropping tools based on machine learning. One portrait cropping tool that frames the faces in an image, and one straighten out function.

3.1 Portrait cropping

The portrait cropping tool frame faces and crops the rest of the image.

3.1.1 My portrait cropping

Figure 6 is the original image that was portrait cropped. Figure 7 is after the image was cropped.
*Figure 8* is the original image. Notice how *figure 9* manages to keep both the faces in the frame.

![Figure 8: Original double persons.](image)

![Figure 9: Cropped image of double persons.](image)

*Figure 10* is a original image. *Figure 11* does have a hard time to keep the ball in the frame. It does frame the face in the image though.

![Figure 10: Original ball person](image)

![Figure 11: Cropped image of ball person](image)
Figure 12 is a original image. Figure 13 does crop of a bit of the girls hand

Figure 12: Original double happy persons

Figure 13: Cropped image of double happy persons

3.2 Straighten out function

The straighten out function does straighten out images if the rotations of the faces are in the same direction. Figures 18, 20 and 22 are original images. These are the images that the different software are going to straighten out.

3.2.1 My straighten out function

Figure 19 highlights how the straighten out function manages to straighten the horizon.
Figure 14: Original person on beach.

Figure 15: Result of the straightening algorithm of person on beach.

*Figure 21* is straighten out, notice how the persons face is completely straight.

Figure 16: Original person with beard.

Figure 17: Result of the straightening algorithm of person with beard.

*Figure 22* is an image that doesn’t need to be straighten out because it is already straight. *Figure 23* shows the flaws of this thesis straighten out function.
Figure 18: Original straight person.

Figure 19: Result of the straightening algorithm of straight person.
4 Analyses and Discussion

4.1 Result

The goal of this thesis was to discover how machine learning could be used to implement cropping tools. Both a functional portrait cropping tool and a straighten out function have been implemented, so the project can be seen as a success. The portrait cropping does a good job with framing the faces in the image and at the same time take some aesthetic rules into account. Without face detection it wouldn’t be possible to frame the faces, because there would be no way to calculate where the faces are in the image. The straighten out function does straighten out images regardless if there is one or more faces in the image, as long as they have the same orientation.

4.2 Challenges

4.2.1 Subjective thinking

One of the big challenges about this project was the aesthetically pleasing part of the cropping. The way people think about images is subjective. So to create and implement a generic algorithm to suit the most people was a challenge. The good thing is that there is some guidelines to follow to get an image to look "popping".

4.2.2 Layout

The layout in the application had specific dimensions. The images that got cropped needed to have the same ratio. If too much was to be cropped in one direction (vertical or horizontal) the image would have been stretched and looked disoriented. It was a challenge to keep the same ratio but at the same time crop so the image looked aesthetically pleasing.

4.3 Restrictions

4.3.1 Portrait cropping

Because this automatic cropping tools are based on information from faces, it is not possible to crop images that don’t include faces. The face recognition algorithm does not in all cases recognize a face if it is too small, which can result in a face being cropped in the middle. There is no way to include things in the image that is important. As can be seen in figure 11, the ball the person is spinning is cropped. This is a relative bad thing when that ball can be the focus point and be making the whole image. Figure 11 does nevertheless frame that face of the image in its cropping.

4.3.2 Straight out function

The straight out function doesn’t take the aesthetically pleasing into account if the rotation is too big. The bigger the rotation is, the more of the image will be cropped. If the aesthetically pleasing would be taken into account, there would be a risk that too much of the image would be cropped which in turn would resolve in a too small image. A too small image would mean it would have to be stretched to fit the display layout. As in portrait cropping, the straight out function is based on faces. So if there is no face in the image, the straight out function doesn’t work.
4.4 Related work

4.4.1 Imaggas application programming interface

This is the result after cropping with Imaggas portrait cropping function. Figure 20 is cropped a little bit on the top and bottom. Figure 21 is cropped in a squared size and does not have any limbs cropped of the persons in the image. Notice how the person in figure 23 still has his ball in the image. Also, in figure 22 are the persons in the image framed with no limbs cut off.

Figure 20: Imaggas cropped image of single person.

Figure 21: Imaggas cropped image of double persons.
4.4.2 Comparing portrait croppings

One thing this thesis and Imagga’s portrait cropping does have in common is that both functions rely on facial recognition. That is the bread and butter of this thesis portrait cropping function. Figure 7 is an example of when the image has been portrait cropped. It does include the whole face of the person. It also tries to align the eyes of the person on the first grid of the “rule of the thirds” to make the image look more appealing. Figure 20 is the result of Imagga’s cropping. It doesn’t crop as much at all of the image. Imagga does however keep the ball in the image on figure 23. That is a big weakness of this thesis portrait cropping - to not being able to take objects in the image into account. As been mentioned earlier, figure 11 does cut the whole ball outside of the image. Figure 8 is an image that does include two faces. Both functions did a good job preserving the faces in the image and aligning it with the “rule of the thirds” grid. Imagga’s portrait cropping function doesn’t crop as much of the image as this thesis function portrait crop does. Figure 13 does have a notable piece of one of the person’s hand cut off. This is because this thesis portrait cropping estimates a bounding box from the faces. This means in this case, that the hand of the persons is too far away from the persons face. That results in a cropped hand. Imagga’s portrait cropping doesn’t crop any limb from the persons. It does center the persons mass and cuts a bit of the right side. This is a good indication that Imagga does more than only face recognition.
4.4.3 Adobe Lightroom

*Figure 24* is not rotated at all. *Figure 25* is straighten out.

*Figure 24*: Adobe Lightrooms result of the straightening algorithm of straight person.

*Figure 25*: Adobe Lightrooms result of the straightening algorithm of person on beach.

*Figure 26* is still crooked. It is rotated a bit too much.
4.4.4 iPhones straighten out application

*Figure 27* is a little bit straighten out. *Figure 28* is completely straighten out.

---

**Figure 26:** Adobe Lightrooms result of the straightening algorithm of person with beard.

**Figure 27:** iPhones result of the straightening algorithm of straight person.

**Figure 28:** iPhones result of the straightening algorithm of person on beach.
Figure 29 is not rotated at all.

Figure 29: iPhones result of the straightening algorithm of person with beard.

4.4.5 Comparing straight out functions

Usually when an image is straightened, the user needs to rotate the image until it is straight.

Auto straighten out functions have one thing in common, they need a tilted line in the image to correct the rotation. They often look for a line in the horizon and if it is tilted they will rotate the image until the line is horizontal and then crop the image so it gets straight. Figure 14 is a good example of an image that got a slanting horizon. All the software succeeded on straighten out that image as can been seen in figures 15, 25 and 28. This thesis straighten out function finds the tilted line with help of face detection. This is not the most efficient way because it can not straighten out an image if there are several people in the image that are rotated in different directions. So even if an image would be tilted, it would not correct it in that case. However, other straighten out functions that searches for a tilted horizon line can get confused if the image contains several tilted lines or if there are no sharp lines. In that case would a straighten out function based on face detection work better. Figure 16 is an example of an image with no distinct horizon line. And as seen in figures 26 and 29, the images were either not straighten out or they were too much rotated.

Figure 18 is an example of an image that is straight. But the person in the image has its head tilted. This mean this thesis straighten out function will rotated the image until it is straight. It will result in an image being crooked instead, as can been seen in Figure 19. The other straighten out functions doesn’t try to straighten out the image, or does significantly straighten out the image.

4.5 Future

4.5.1 Portrait cropping

One thing that would improve the portrait cropping tool would be if the user could adjust the dimensions of the cropping. So instead of the portrait cropping, cropping the images instantly. It would rather display a suggestion to the user how it could be cropped. And then the users could move the corners into what they would prefer. This would be a great solution to solve the subjective matter.
4.5.2 Straighten out function

As the same as portrait cropping, the straighten out function would be improved if the user could manually adjust the rotation. The straighten out function would rotate the image automatically and then would the user be able to adjust the last bit. About the same way as the iPhone's casual editor application works.

Figure 30: Portrait cropping with advise.
5 Conclusion

Is machine learning a good combination with automatic cropping tools? It does a good job! Is machine learning a good solution to subjective problems? Maybe. It may be a step in the right direction. Before taking on this task, I knew the subjective thinking part would be a problem for the portrait cropping and straighten out function. The solution for the portrait cropping was to make a general case that would work for the most cases. Even though it might not always crop as the user intend to, it may give the user inspiration. And if manual adjustable cropping slides would be implemented, the auto portrait cropping would be a great guideline. The portrait cropping nevertheless lives up to it’s name. And if it is a portrait crop the user wants, they are going to have a portrait that succeeds to put in aesthetics into the image. The straighten out function didn’t have to tackle the subjectively thinking problem the same way, if an image is oblique, it is oblique. However, the subjective thinking problem still applies. The more angle the faces have in the image, the more is going to get cropped away. So sometimes it would be better to have the face/faces remain a little bit tilted to spare the size of the image and sometimes it is ”artsy” to have the image a little bit tilted. Face recognition is a good supplement, but it does need something more. Something as object recognition, something that can recognize the outline of an object. That way it would elude cropped limbs. Imagga does a good job with creating a bounding box around the persons bodies. Machine learning is a powerful tool that doesn’t need to be implemented for everything, but can be implemented for everything. It does tend to make things more humane. But will it be able to predict exactly how a person’s subjective thought is?
6 References

References


