HOCHSCHULE
RAVENSBURG-WEINGARTEN
UNIVERSITY
OF APPLIED SCIENCES

COMPUTER VISION

COMPULSORY PROJECTS

www.rwu.de
info@rwu.de

GENERAL RULES AND INFORMATION

Compulsory Projects

The reports

In these slides, you will get a collection of all information needed to solve the tasks of the Hands-On projects.

For each task, you will have to **hand in a report via Moodle with a strict deadline**. In this report you will have to answer all questions defined in the task descriptions.

All limitations of your reports (e.g. page-count) will also be published via Moodle.

Compulsory Projects
The reports

A report should always carry all information for the reader to reproduce your algorithms to get the same results as you. At least provide always the following:

- **What?**

  What is your answer or result to the question?

- **How?**

  How did you come to your conclusion? How did you calculate your result?

- **Why?**

  Why did you use your approach? Why did you define the function the ways you did it?

Compulsory Projects
The reports

A report should always carry all information for the reader to reproduce your algorithms to get the same results as you. At least provide always the following:

- **What?**
- **How?**
- **Why?**

Describe your own functions in detail to allow a reproduction by the reader. Here, pseudo code or a flow chart can be helpful to describe longer or more complex functions.

Compulsory Projects
The reports

Follow the general rules for a scientific report. Read the “How to cite” guide provided.

Any kind of plagiarism will result in an automatic failing grade (5.0) for all students involved.

Compulsory Projects
The reports

As a typical What You See Is What You Get (WYSIWYG) editing software (e.g. Microsoft Word, Apple Pages) will reach its limitations for a scientific report, **we will use LaTeX only**.

To generate your report, use one of the many available LaTeX editors (e.g. TeXworks): https://en.wikipedia.org/wiki/Comparison_of_TeX_editors

As you will most likely have to use LaTeX for compiling your Master thesis, this is a good place to start practicing.

Compulsory Projects
The reports

There are many templates available:

https://www.latextemplates.com/

We also have RWU templates:

https://fbe-gitlab.hs-weingarten.de/prj-corporate-design/latex-rwustyle

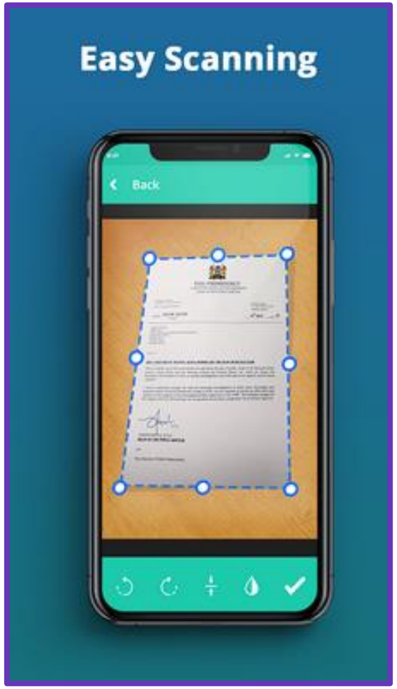
Now, let us talk about what we want to do....

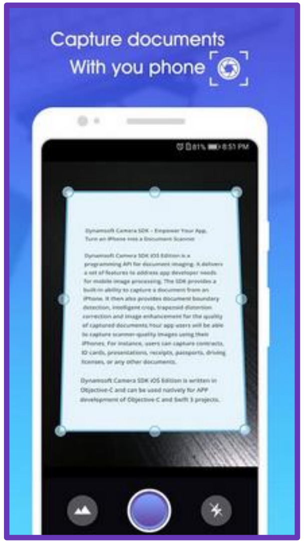
TASK 1:
THE SCANNER

The Scanner
Some examples

A common application of image processing or computer vision is a **document scanner**.

They are typically used on mobile devices. Maybe you have used one already.





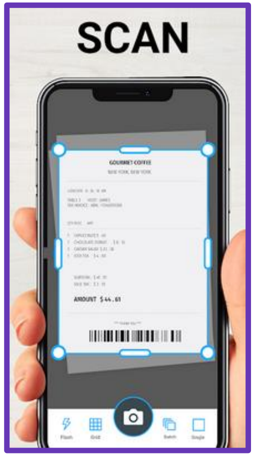


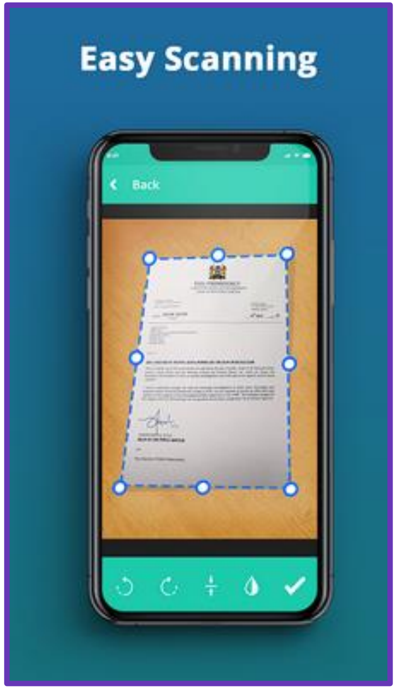
Fig.: Examples of scanners from the App Store Google Play.

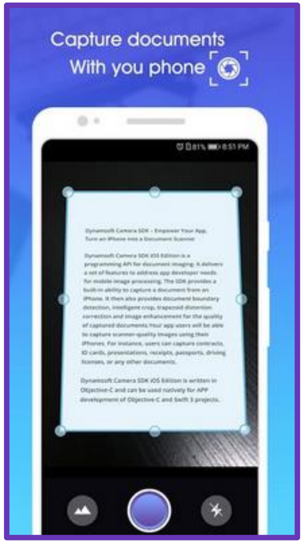
The Scanner

Some examples

To provide a good scan of the document, we need a **perspective transformation**.

The types of transformations are covered in OpenCV as well.





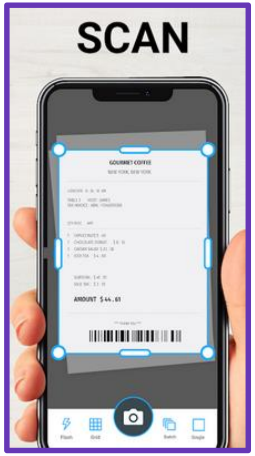
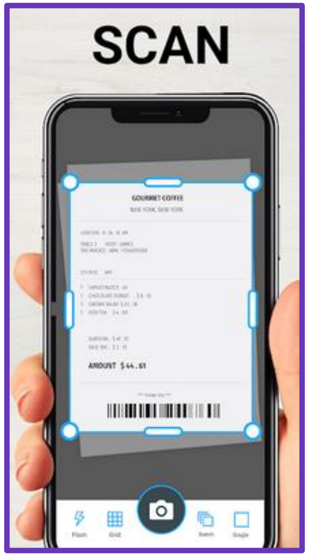


Fig.: Examples of scanners from the App Store Google Play.

The Scanner

Some examples

As we can see, the different applications do have different functionality.



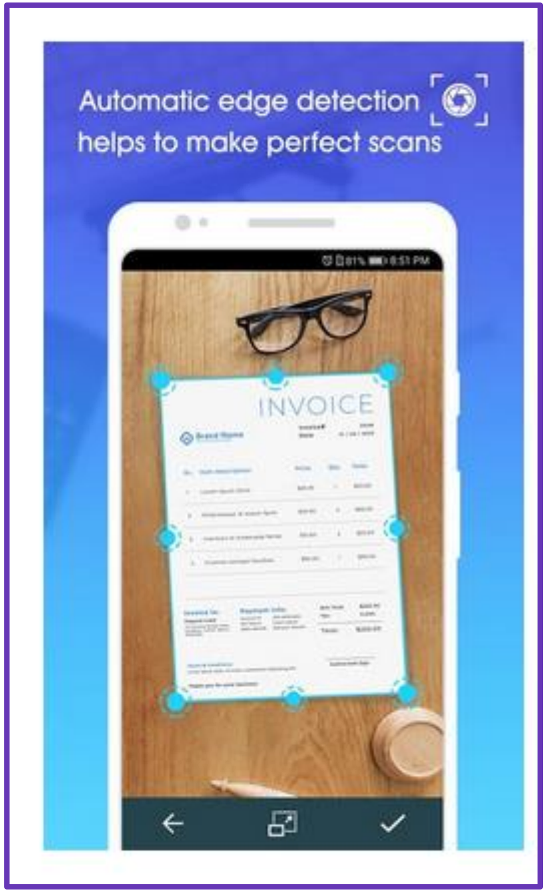
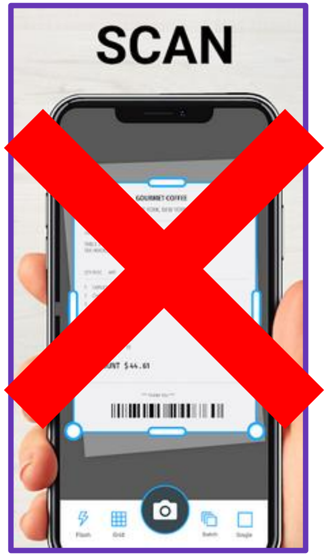


Fig.: Two scanners in comparison. The example above needs manual input.

The Scanner
What we have to do

As we want an automatic document scanner, we want a solution which does **detect the document automatically** and then provides a proper scan of the document.



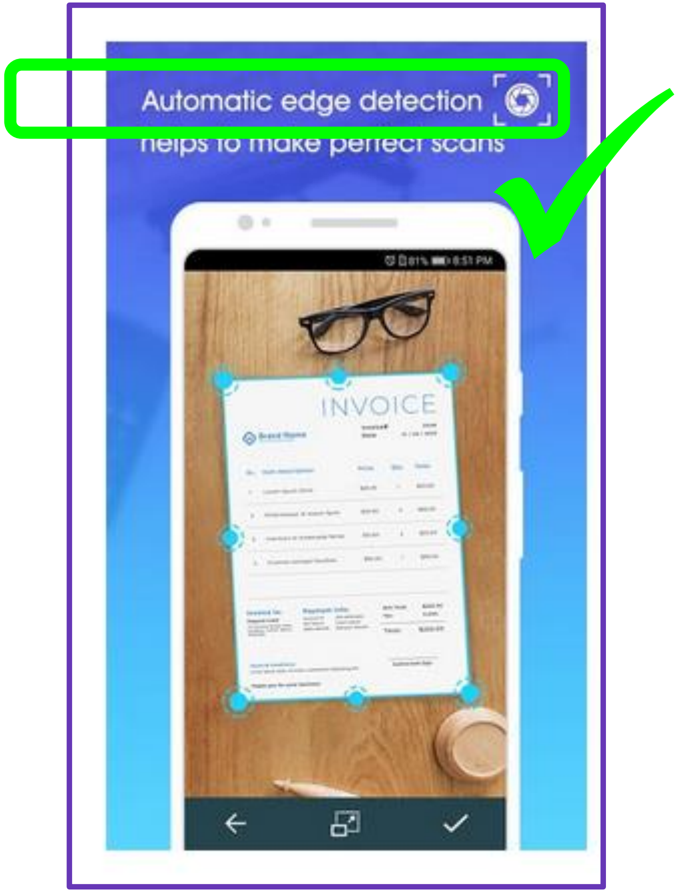
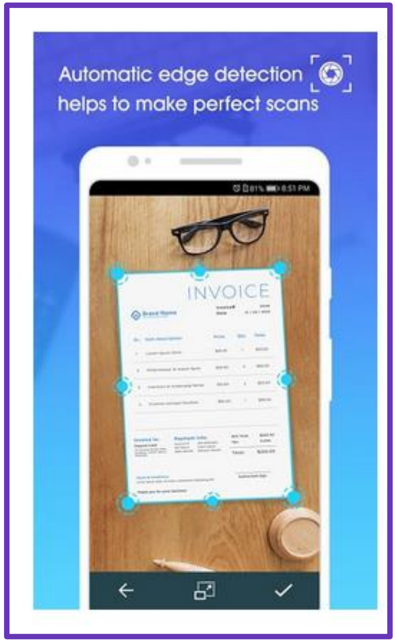


Fig.: Above: scanner without automatic corner point detection. Right: scanner with automatic edge detection.

The Scanner
What we have to do

To limit the possible inputs and allow a better automatic result, we can assume that the document to be scanned is following the international standard **ISO 216** (e.g. DinA4).

To allow a good user-experience, your algorithm has to be able to handle any document (ISO 216) with any rotation.



The Scanner
What we have to do

As a result of our scanner we want a **clean scan in a small data format**.

For this, after the perspective transformation the document has to be transformed into **a binary image** (black and white).

For this step, a simple thresholding approach (as seen on the right) will not be enough.

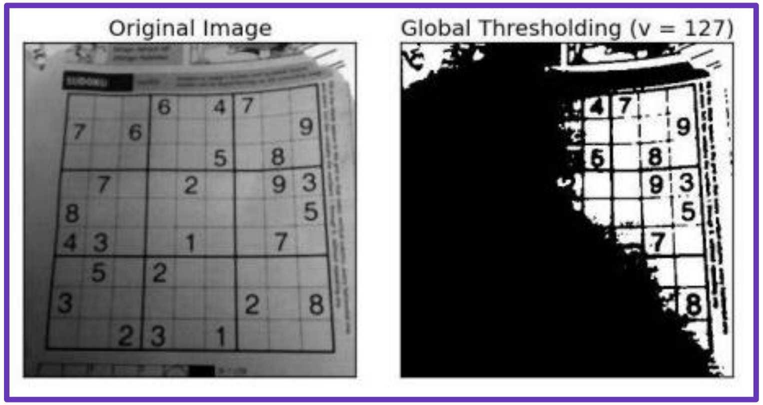


Fig.: Bad result of a simple thresholding. This is exactly what we not want to have.
Source: OpenCV documentation.

Task 1: The Scanner
What we have to do

Goals of task 1:

For our scanner to work, we need to fulfil a three step algorithm:

- Calculate the position of the **four corner points**
- Use a perspective transformation to get a clean **top-view** of the document.
- Filter the scanned document to get a clear **binary** image.

Describe in your report how you did solve this problem, provide all information needed to reproduce your algorithms. Include images where they are helpful to describe your steps.

TASK 2:
T.B.D.

HOCHSCHULE
RAVENSBURG-WEINGARTEN
UNIVERSITY
OF APPLIED SCIENCES

VIEL SPASS UND VIEL ERFOLG!

Doggenriedstraße
88250 Weingarten

Postfach 3022
88216 Weingarten

www.rwu.de
info@rwu.de