Identifying pokémon cards with computer vision

github.com/hugopeixoto/ptcg-detection

Hugo Peixoto (he/him)
hugopeixoto.net

My pokémon

Process overview

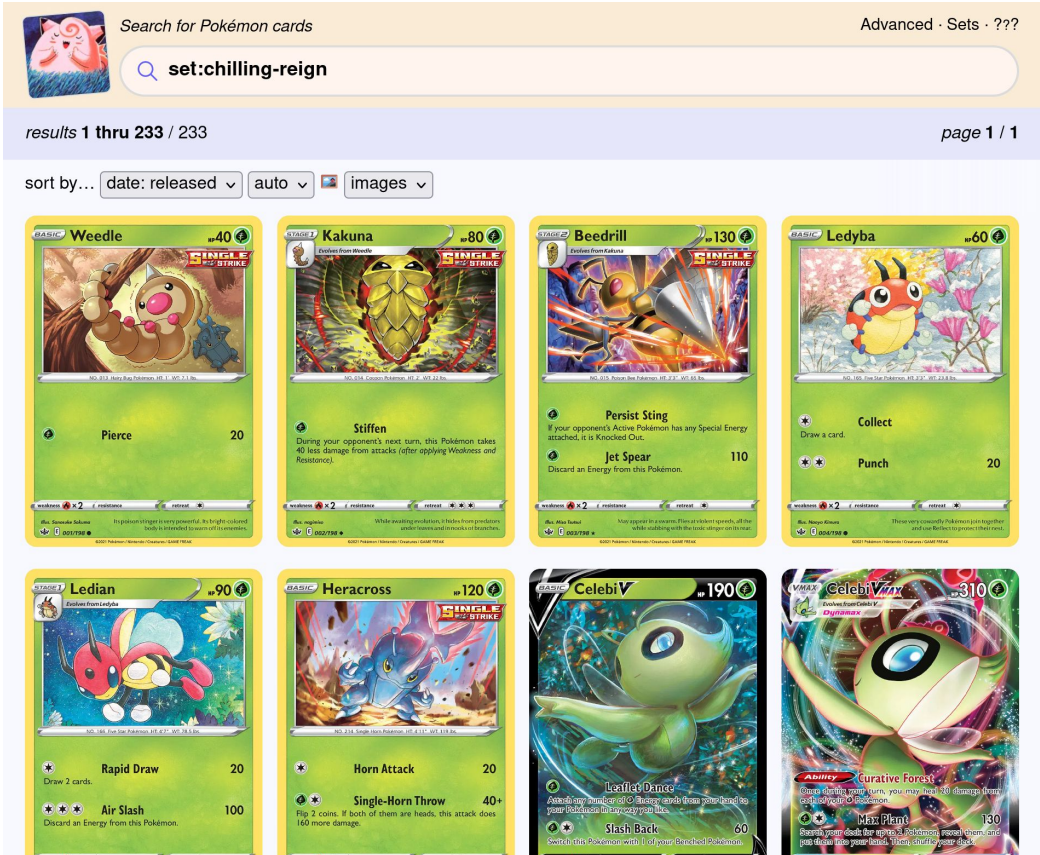
- Grab frame from webcam stream
- Extract the card from frame
- Search our dataset for a similar card

Lowering of expectations

Grabbing a dataset

https://pkmncards.com/

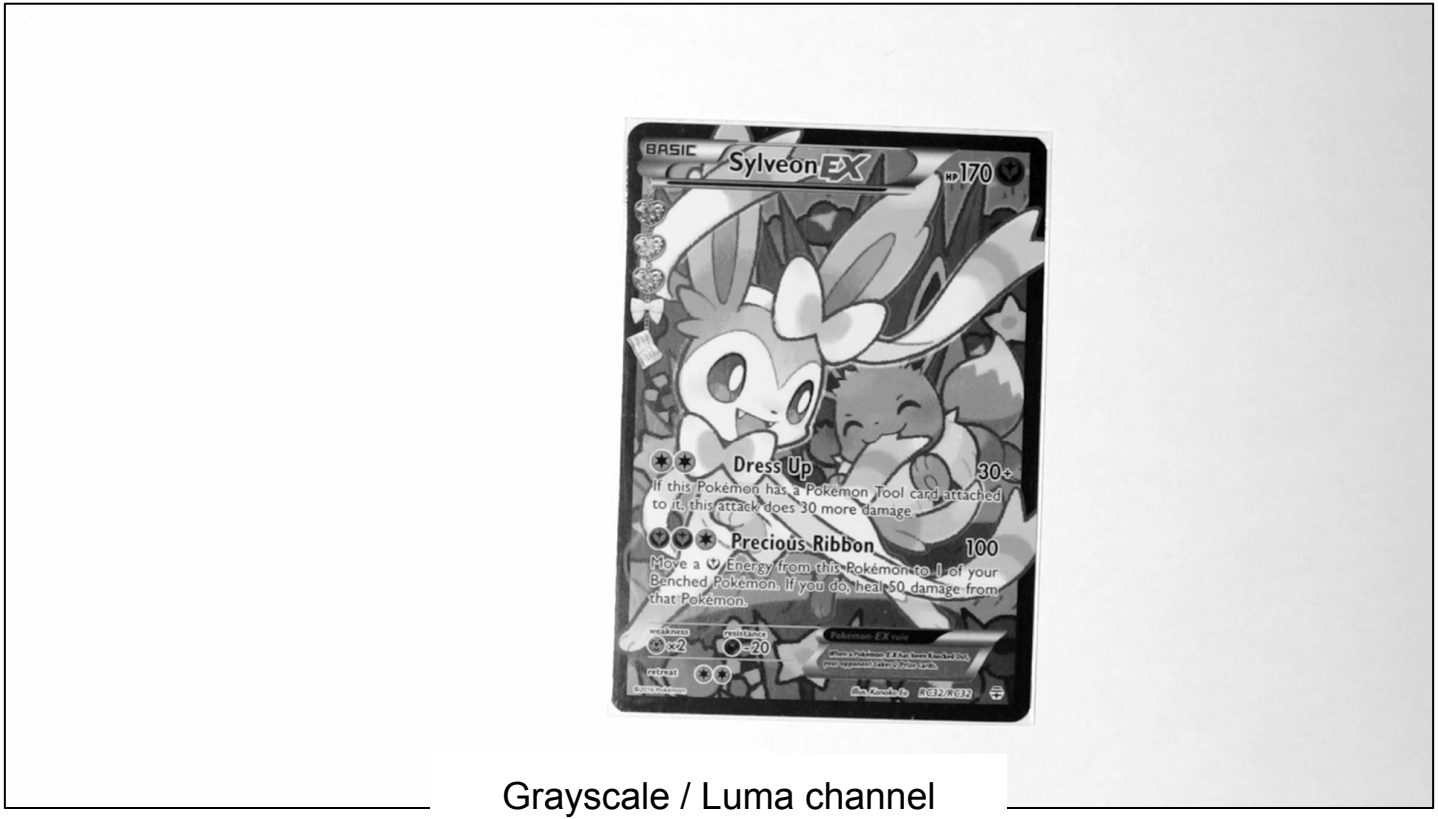
~14 thousand cards



Detecting the card

Detecting the card

Detecting the card



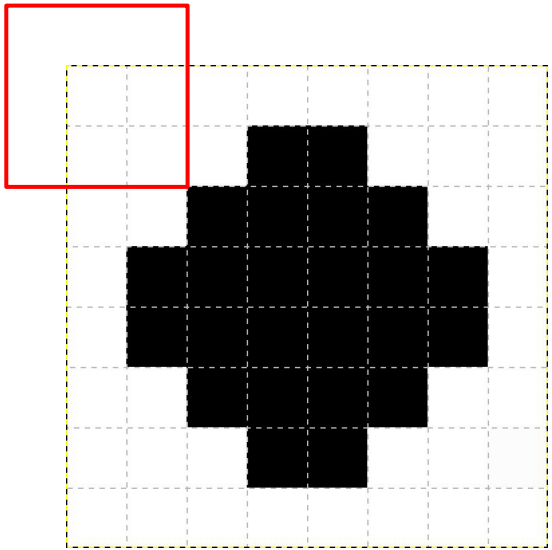
Grayscale / Luma channel

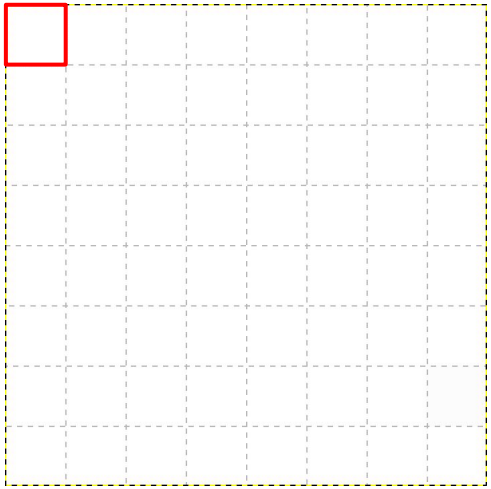
Detecting the card

Detecting the card

Sobel operator

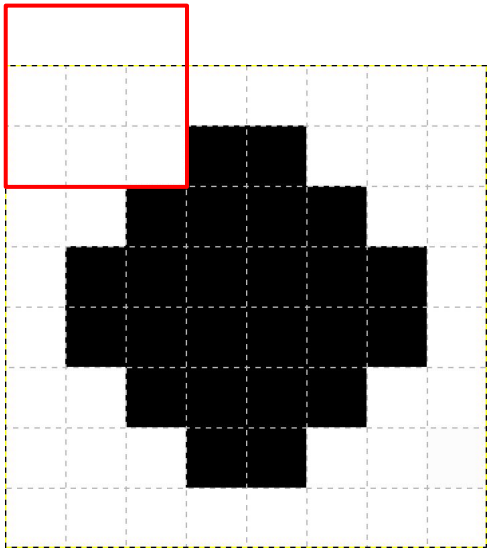
Detecting the card

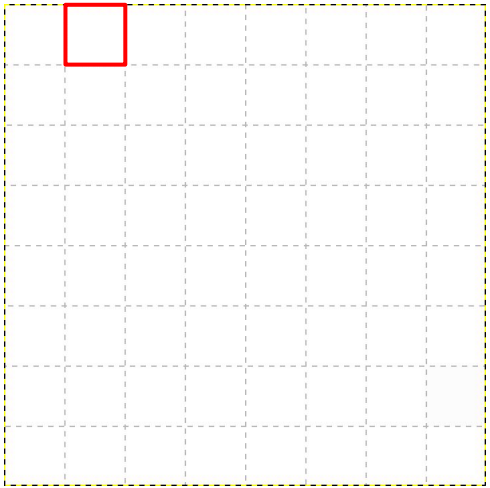




Kernel-based image filters

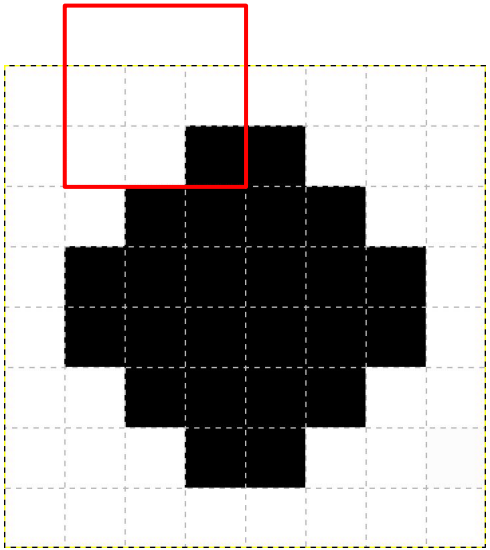
Detecting the card

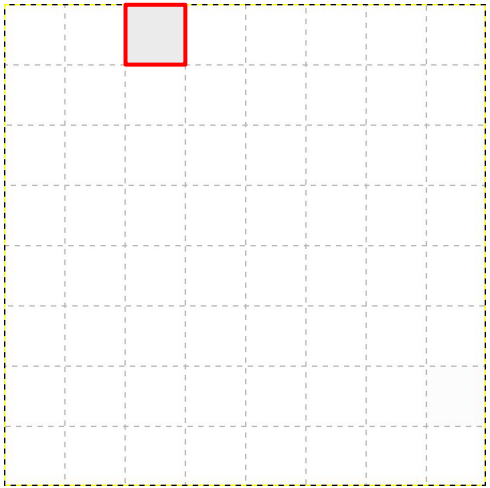




Kernel-based image filters

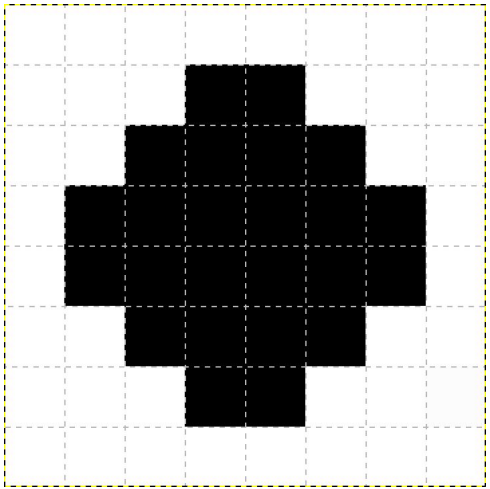
Detecting the card





Kernel-based image filters

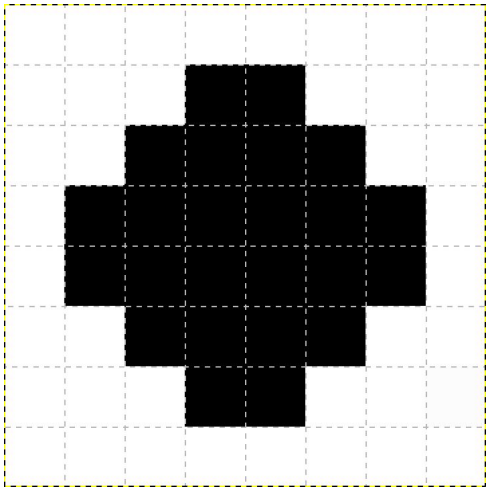
Detecting the card

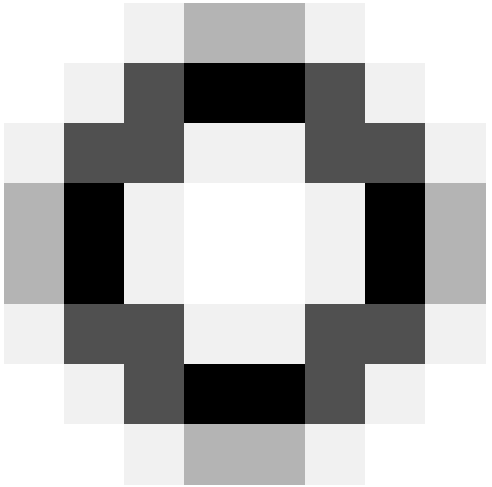


0	0	360	806	806	360	0	0
0	360	1081	1140	1140	1081	360	0
360	1081	1081	360	360	1081	1081	360
806	1140	360	0	0	360	1140	806
806	1140	360	0	0	360	1140	806
360	1081	1081	360	360	1081	1081	360
0	360	1081	1140	1140	1081	360	0
0	0	360	806	806	360	0	0

Sobel operator

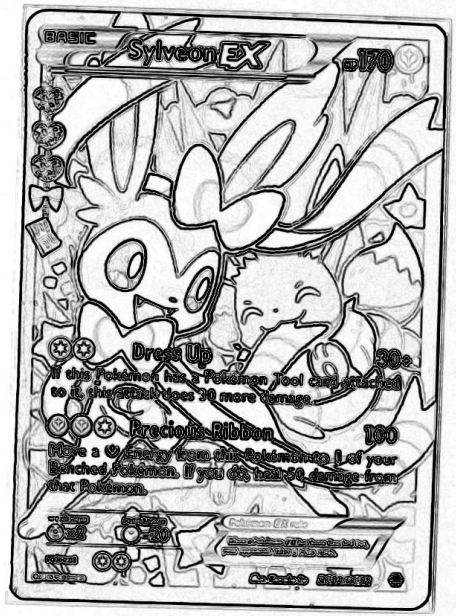
Detecting the card





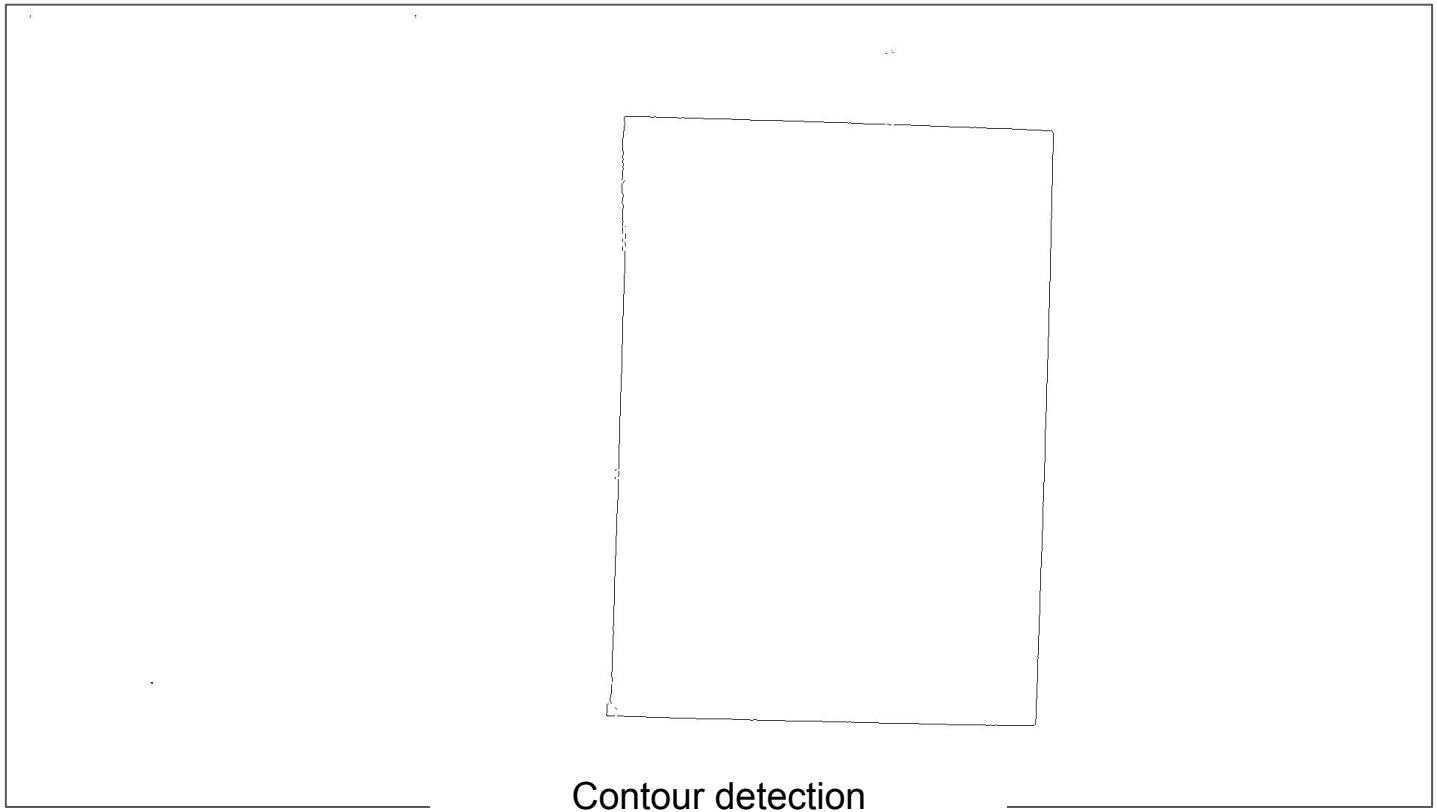
Sobel operator

Detecting the card



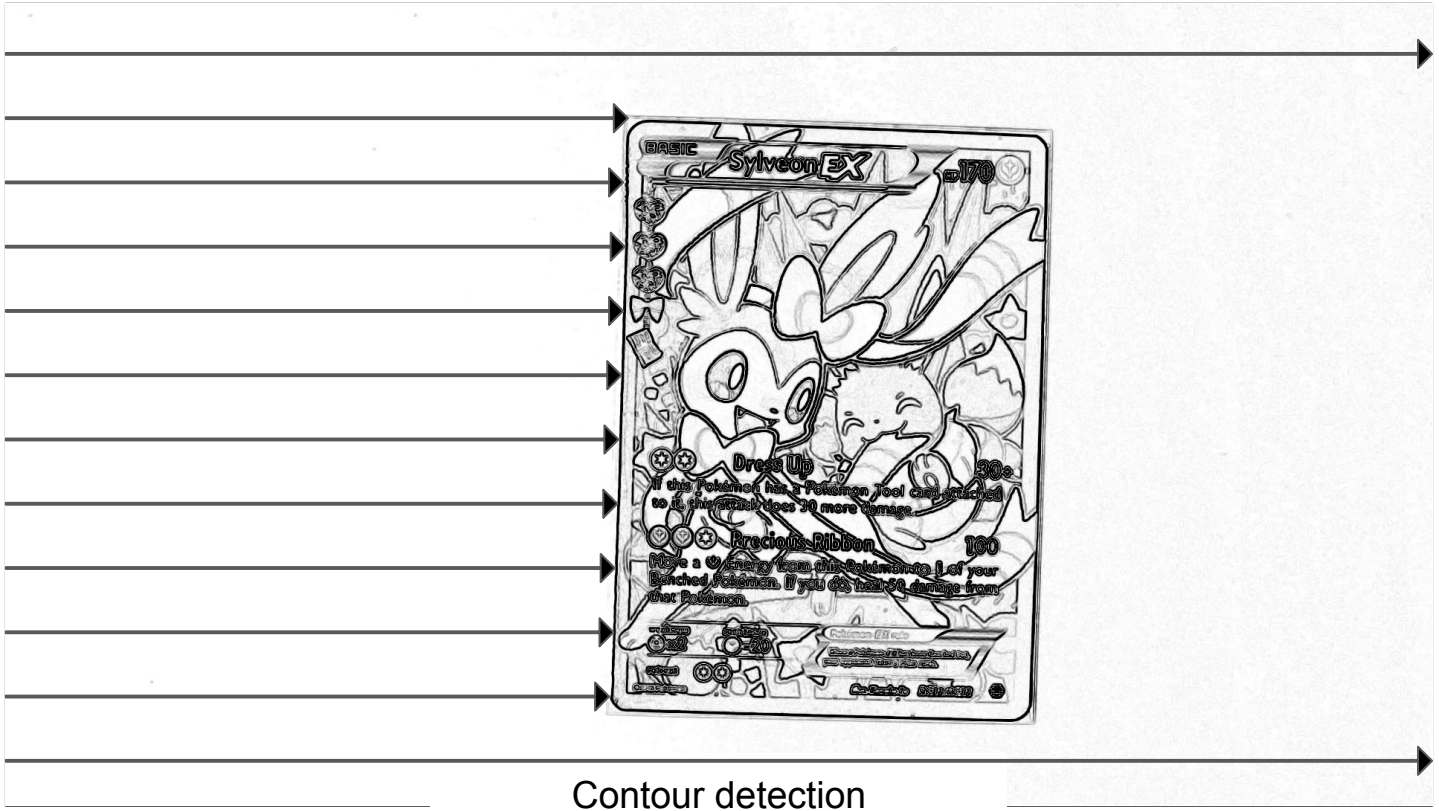
Sobel operator

Detecting the card



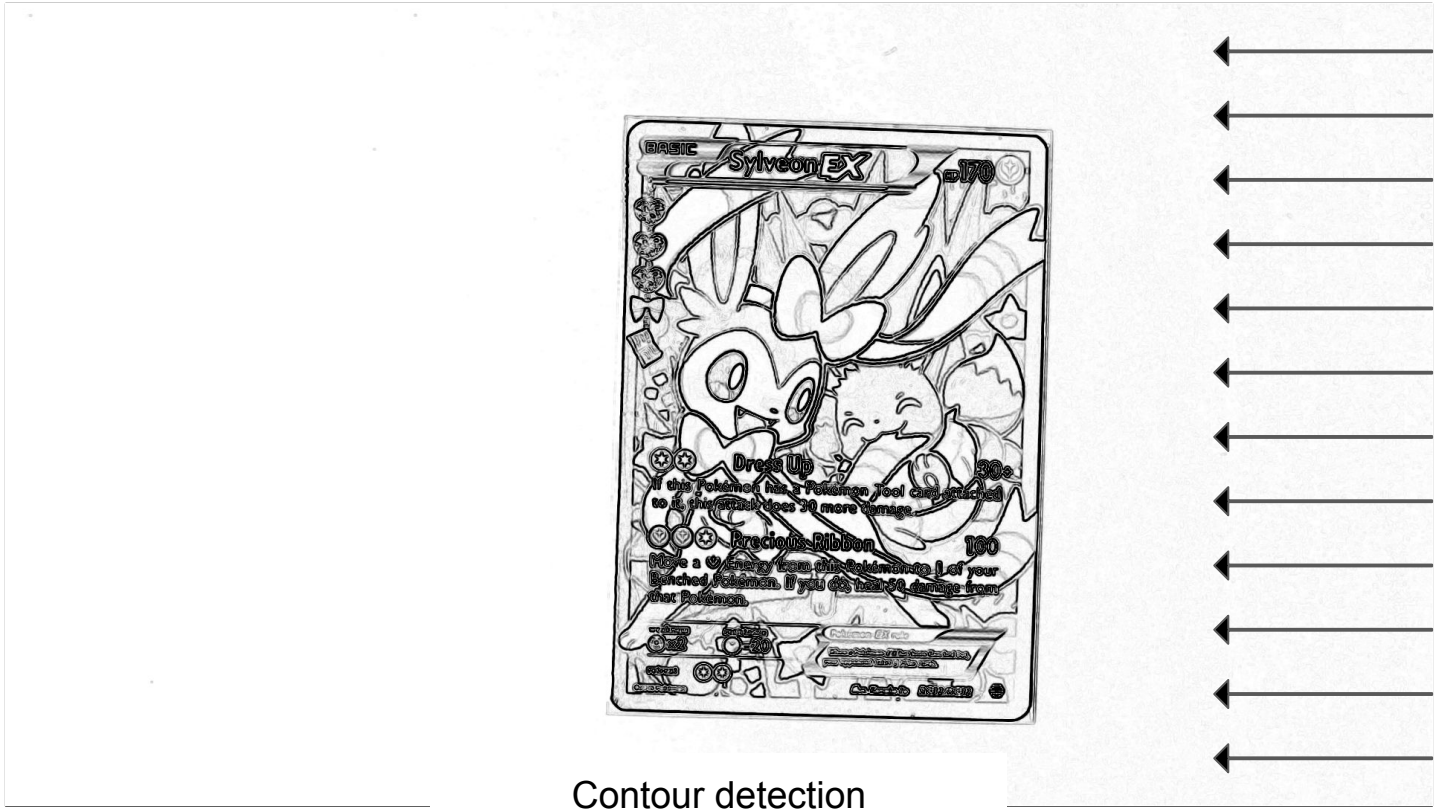
Contour detection

Detecting the card



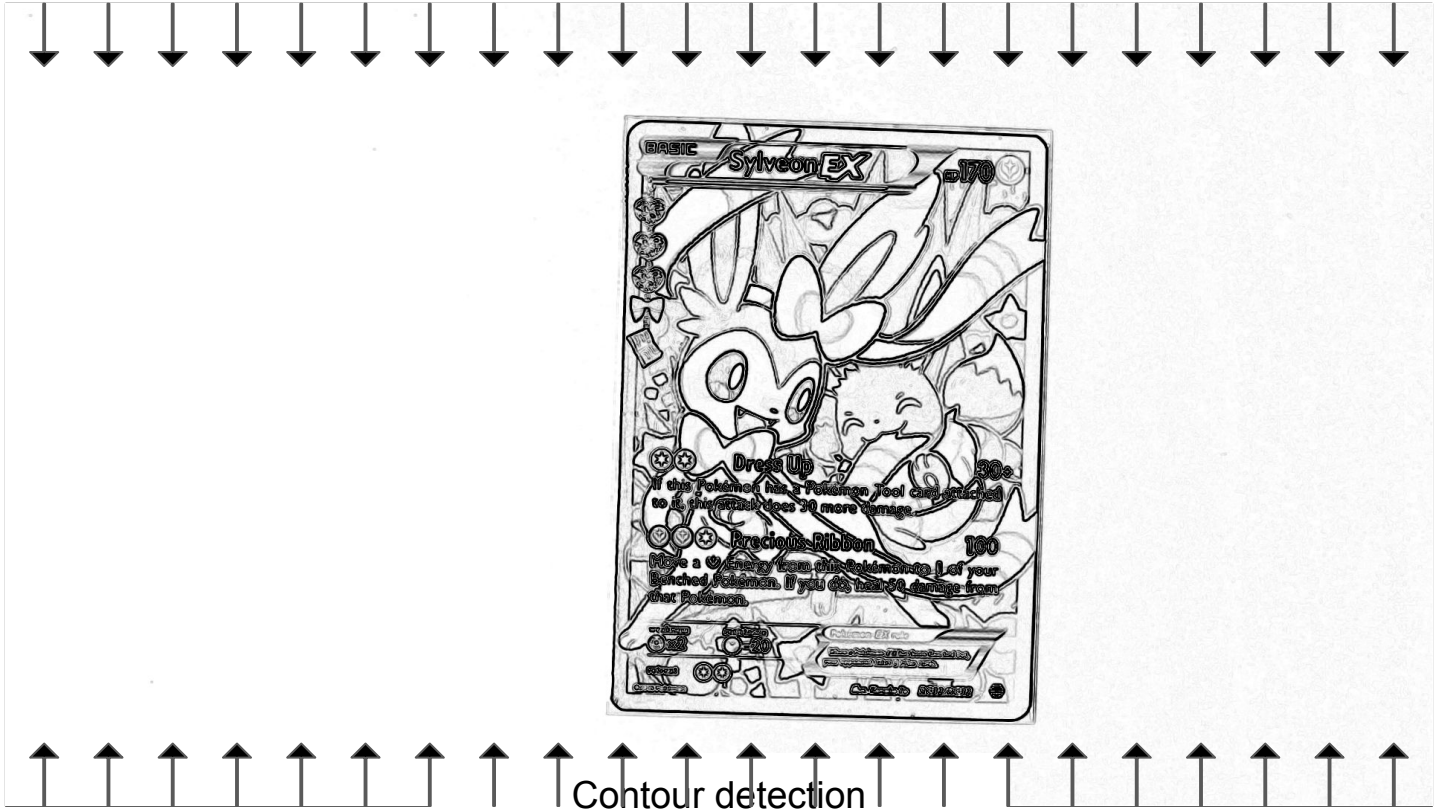
Contour detection

Detecting the card



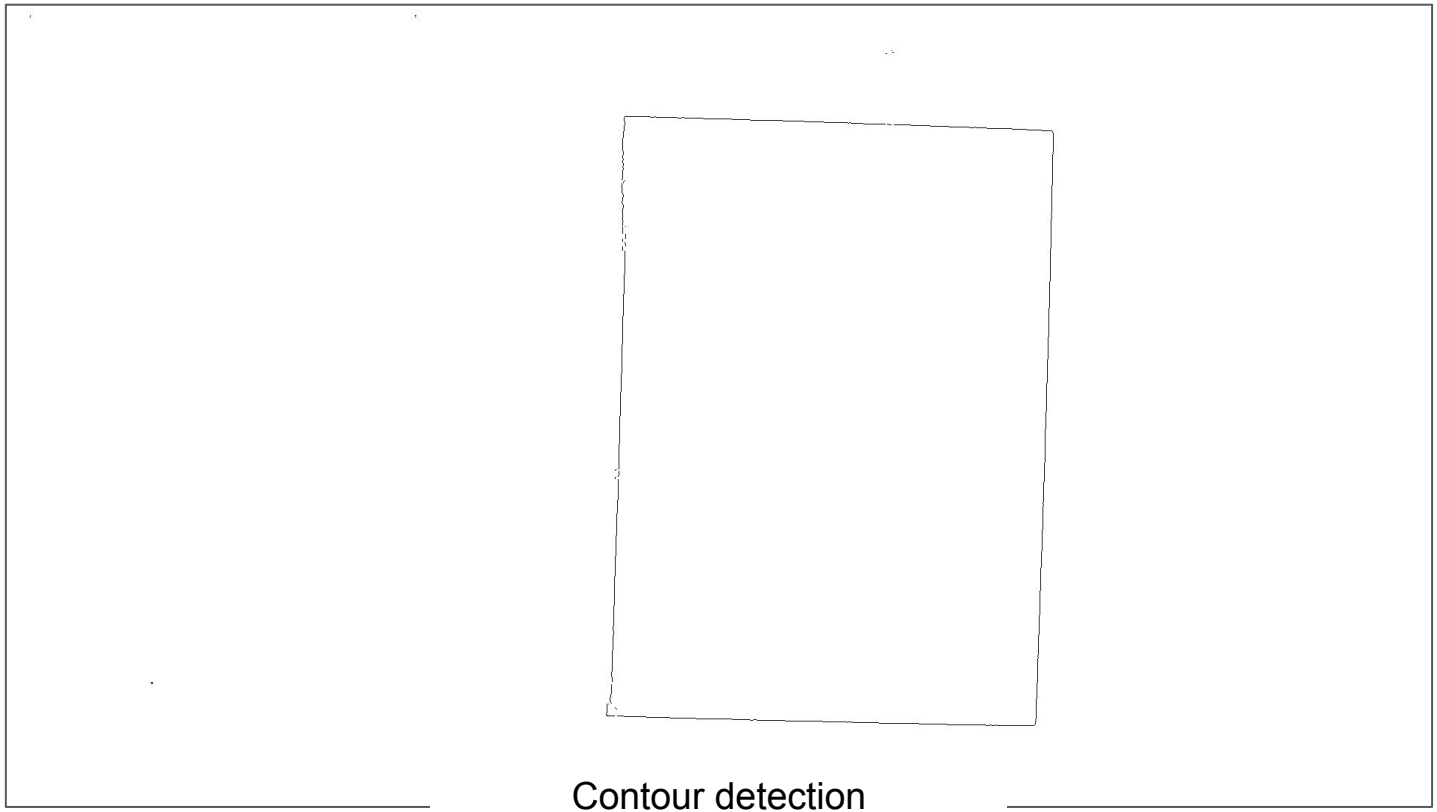
Contour detection

Detecting the card



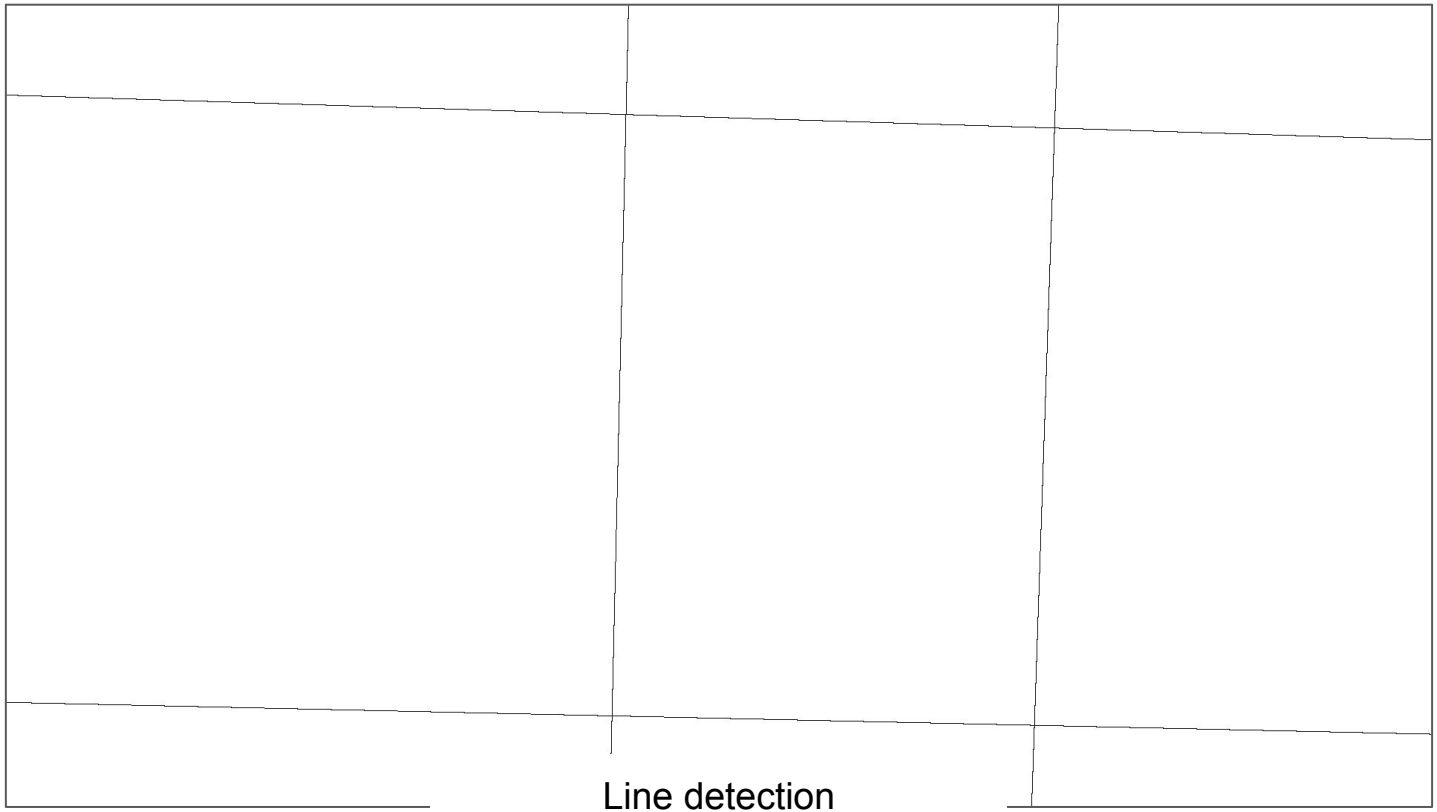
Contour detection

Detecting the card



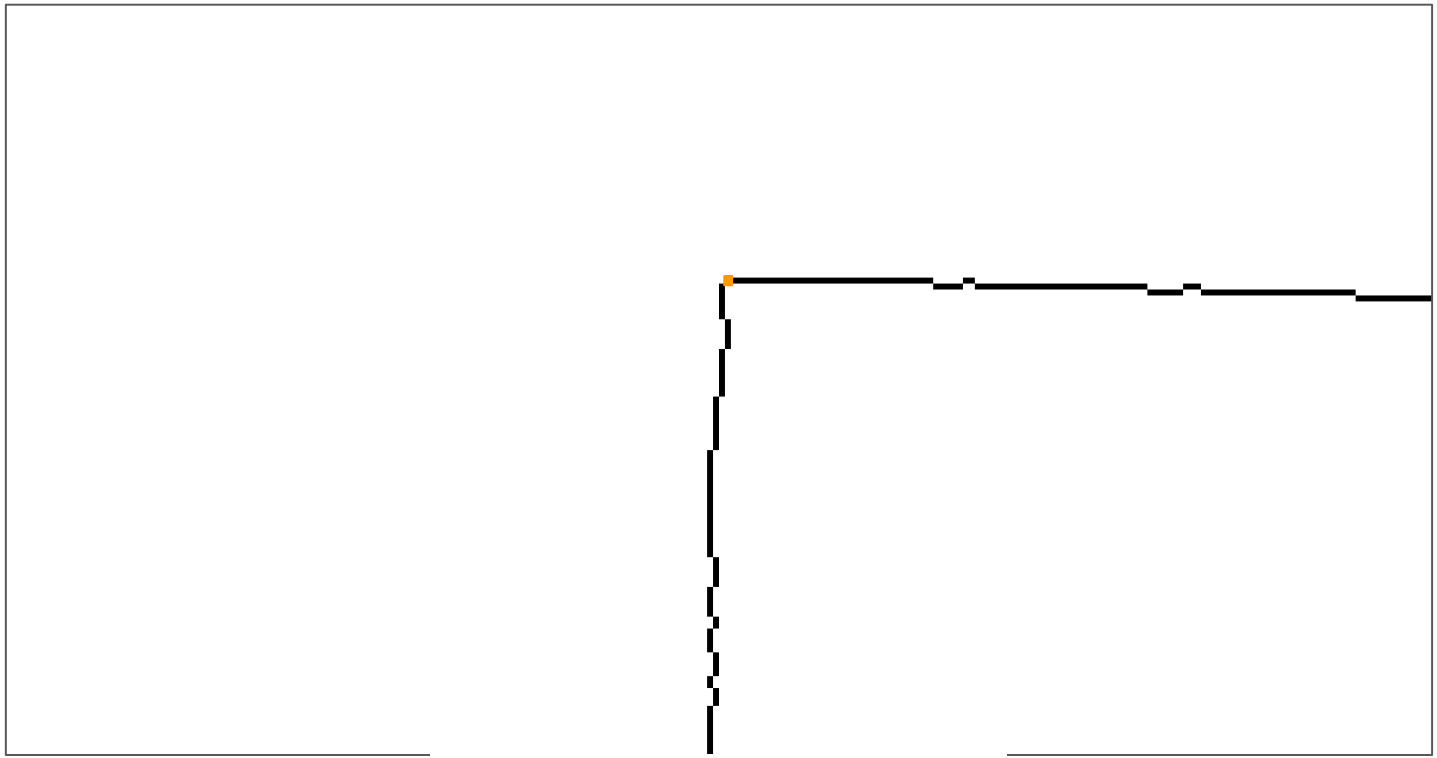
Contour detection

Detecting the card



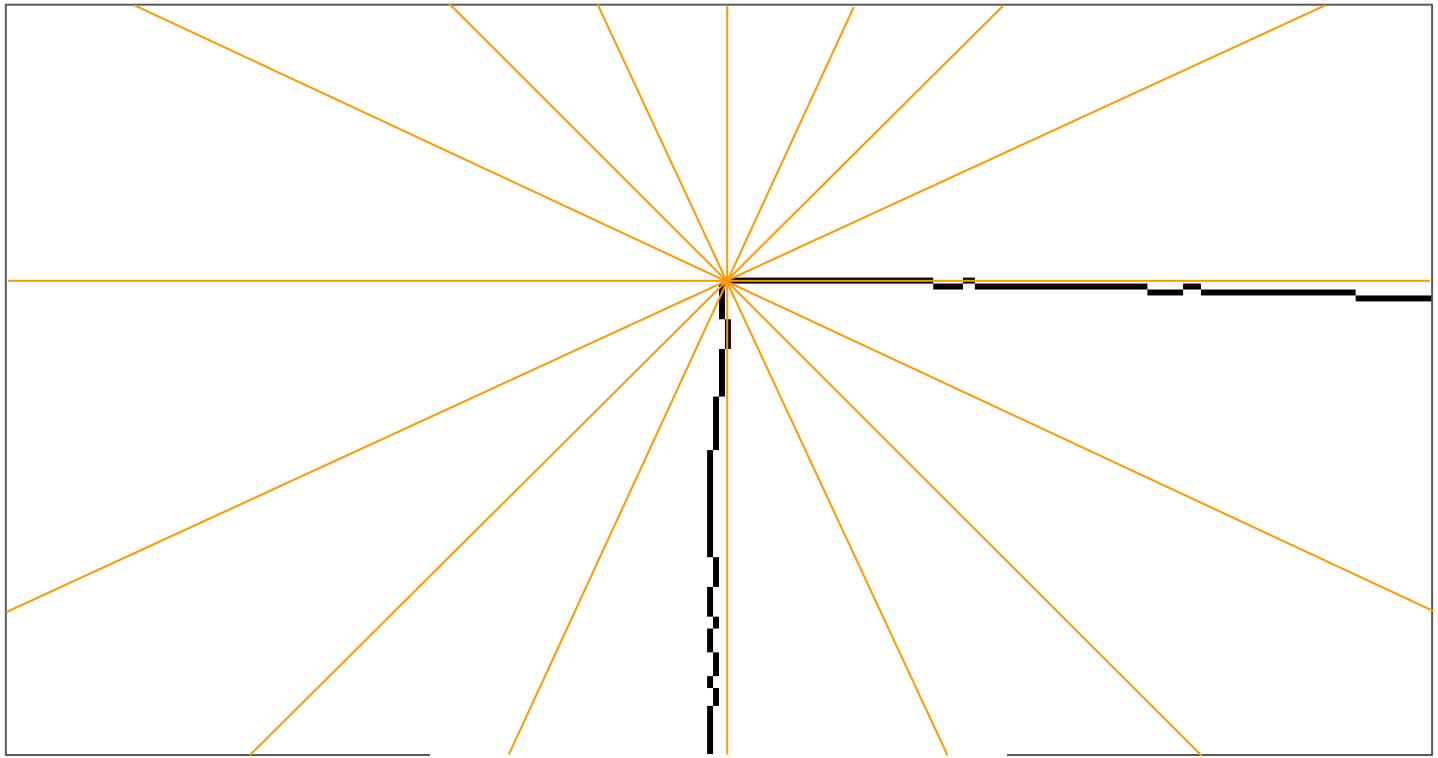
Line detection

Detecting the card



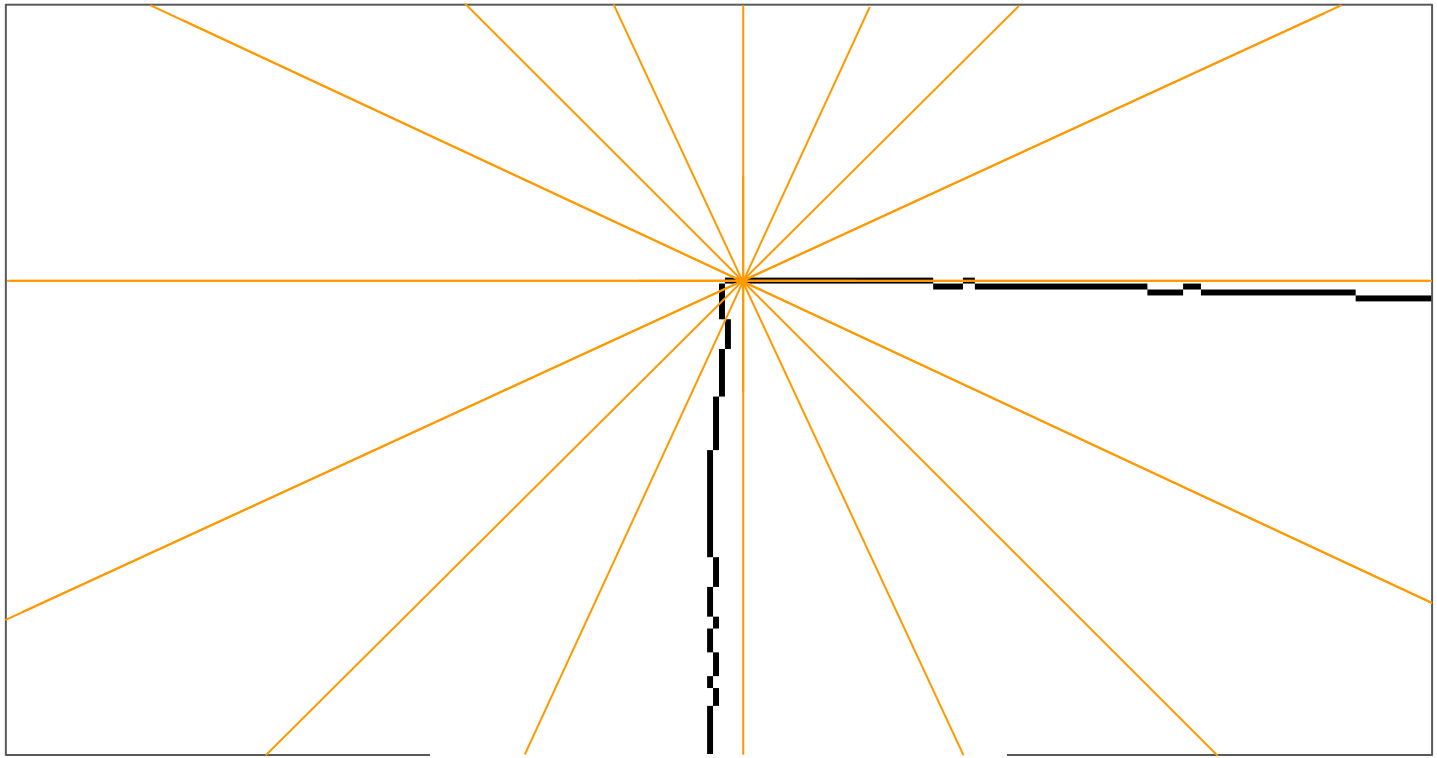
Hough transform

Detecting the card



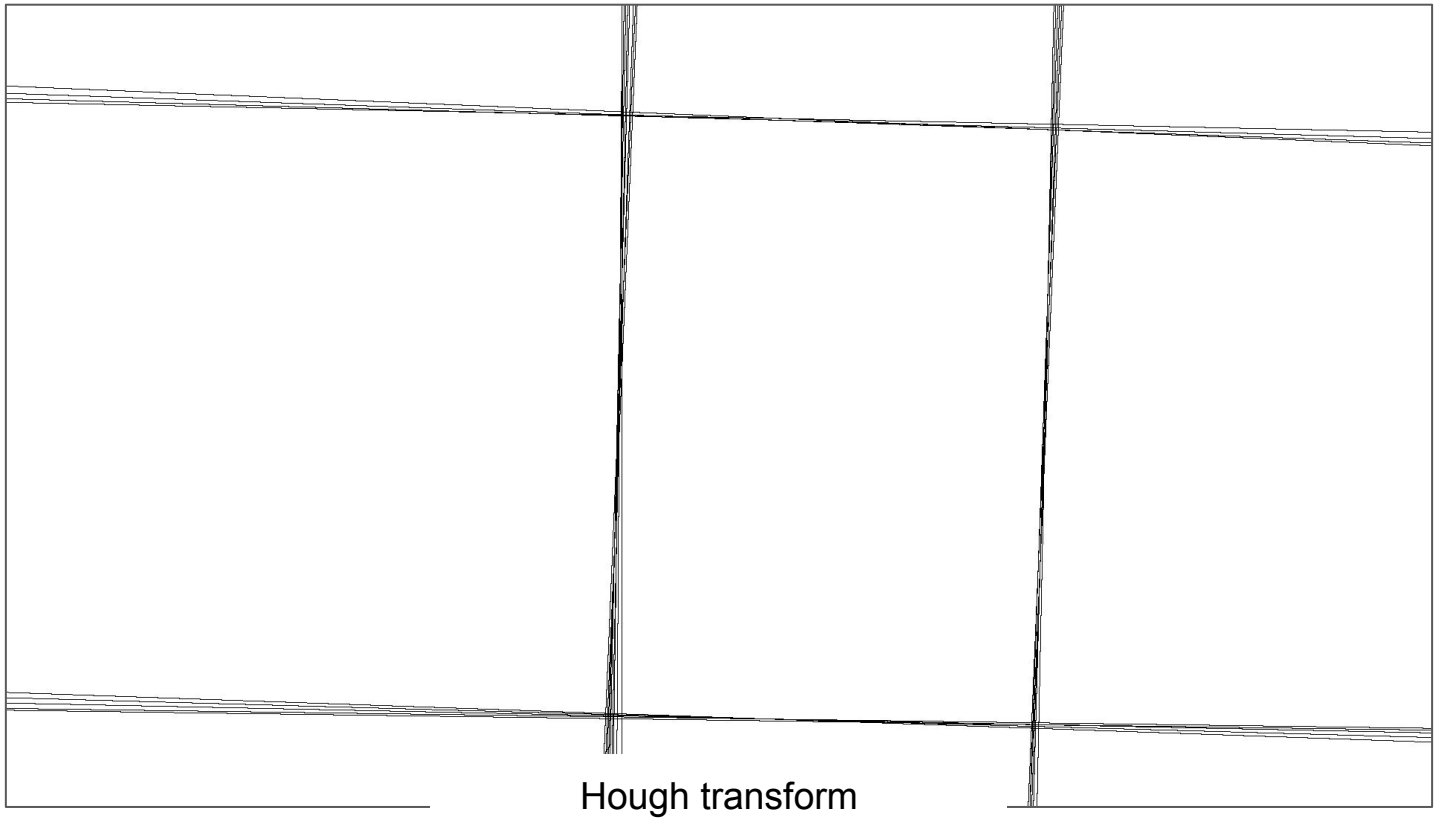
Hough transform

Detecting the card



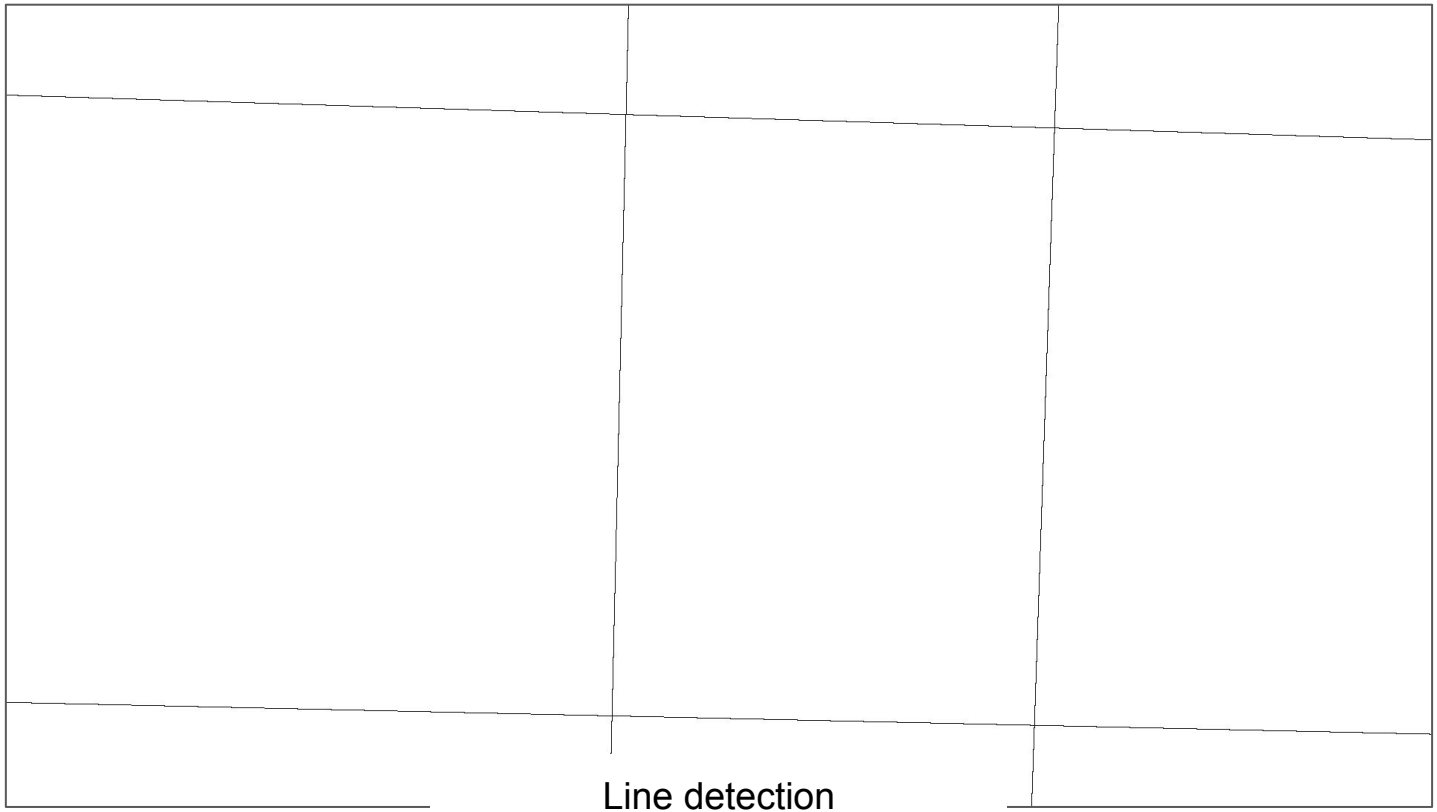
Hough transform

Detecting the card



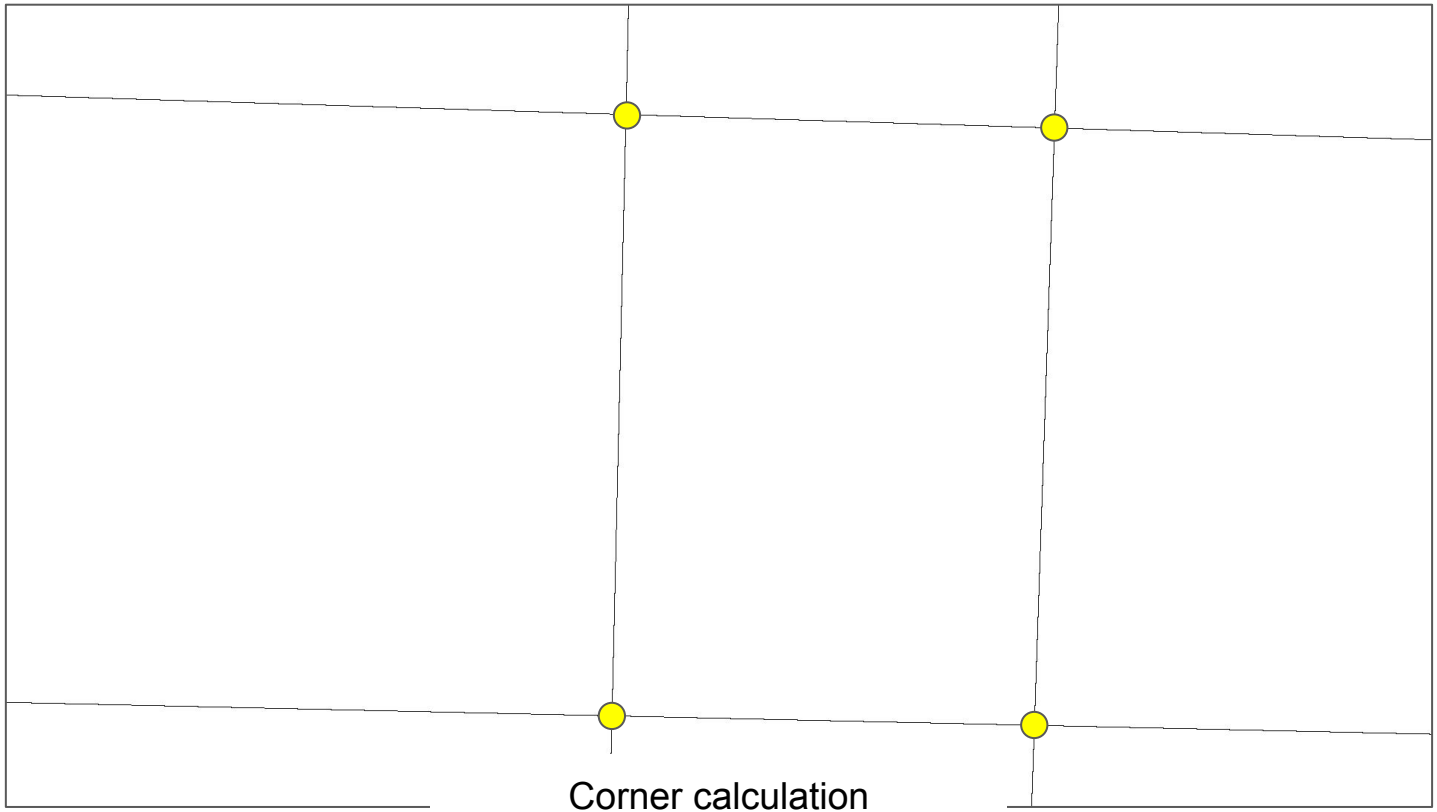
Hough transform

Detecting the card

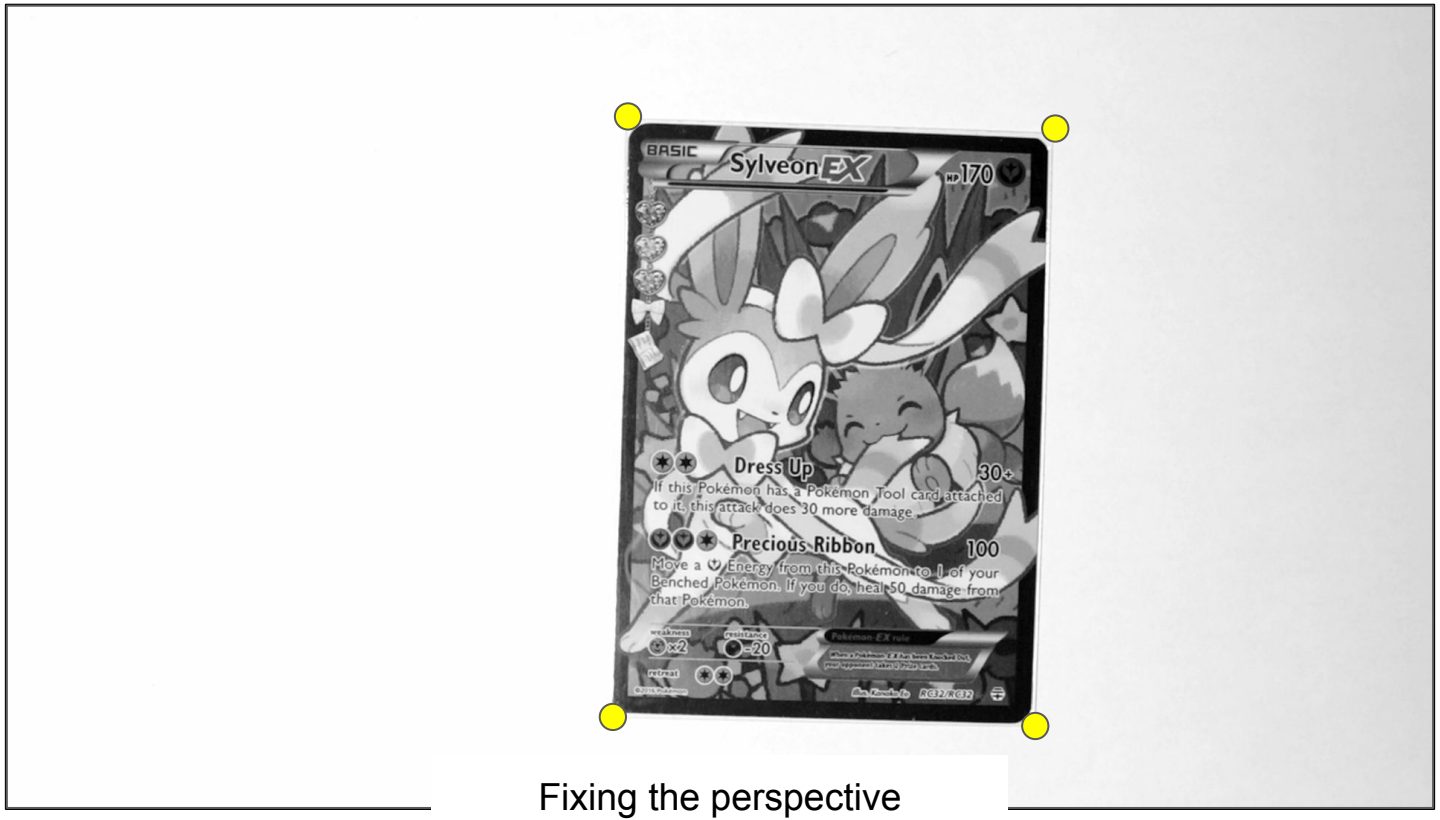


Line detection

Detecting the card



Detecting the card



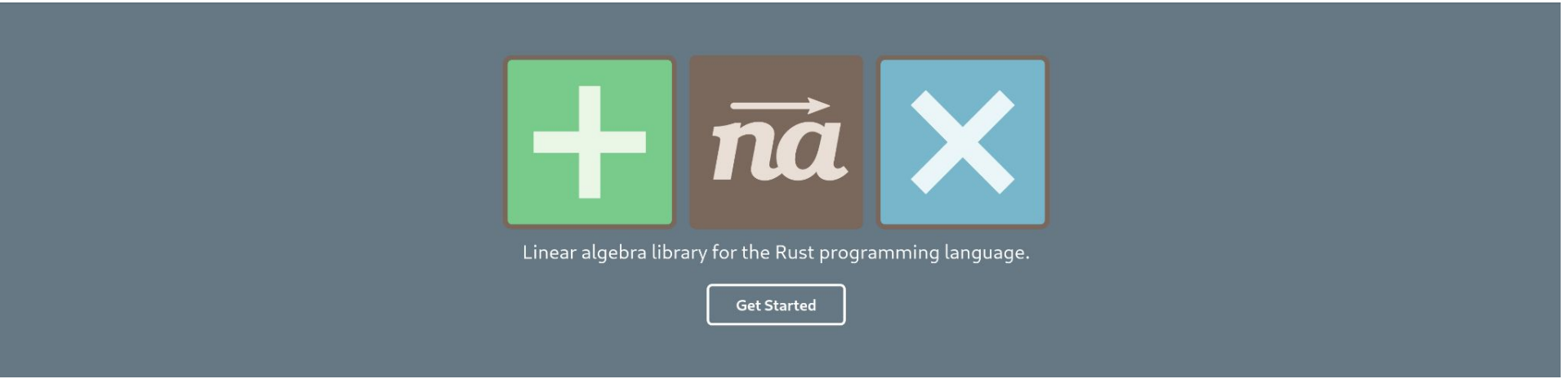
Fixing the perspective

Detecting the card

Fixing the perspective

Detecting the card

nalgebra.org

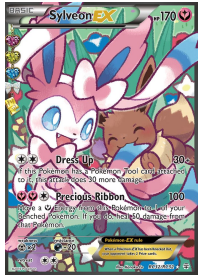


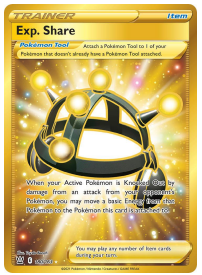
Fixing the perspective

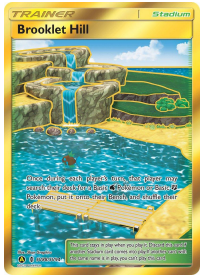
Detecting the card

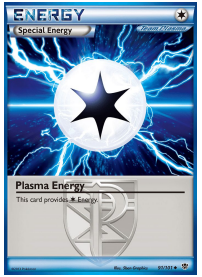
Fixing the perspective

Finding a match

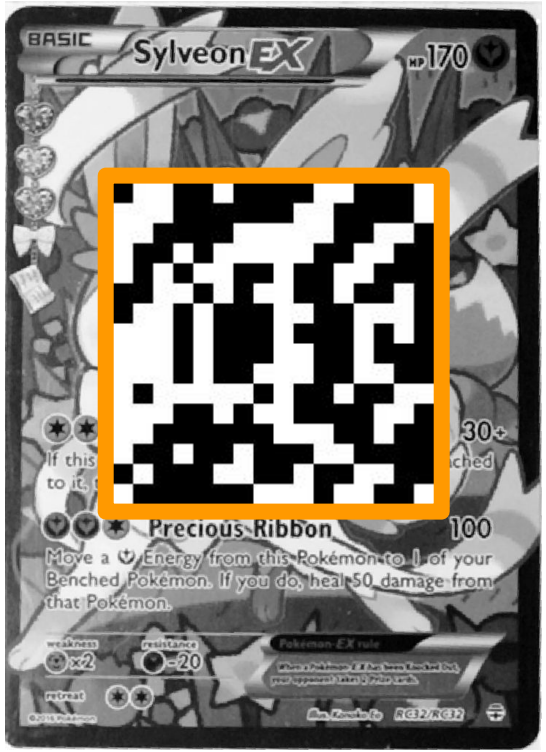








Finding a match



perceptual hashing

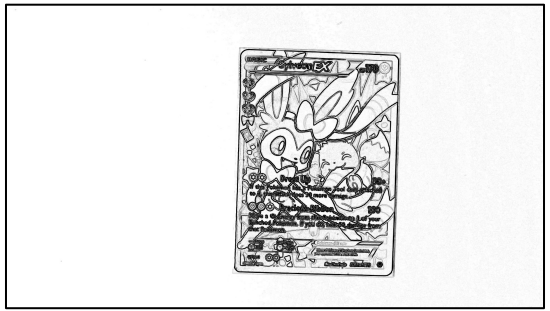
Finding a match

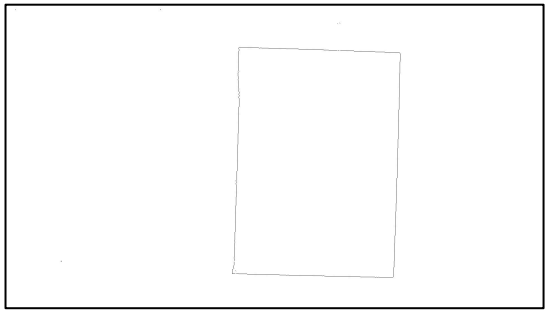
resize

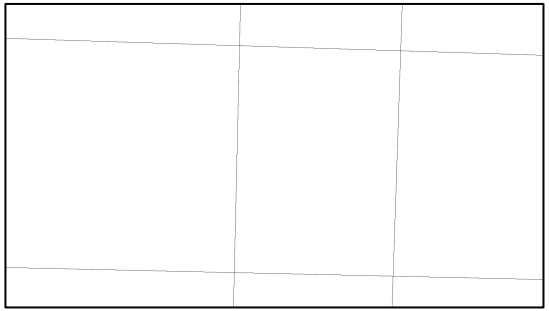
binarize

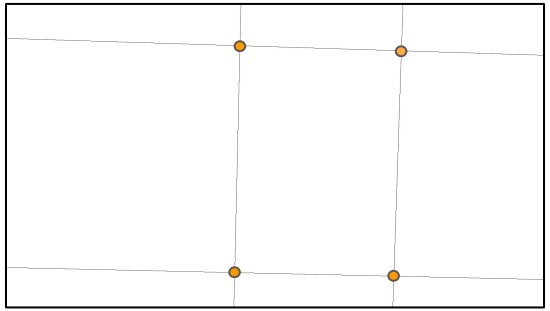
perceptual hashing

Process overview

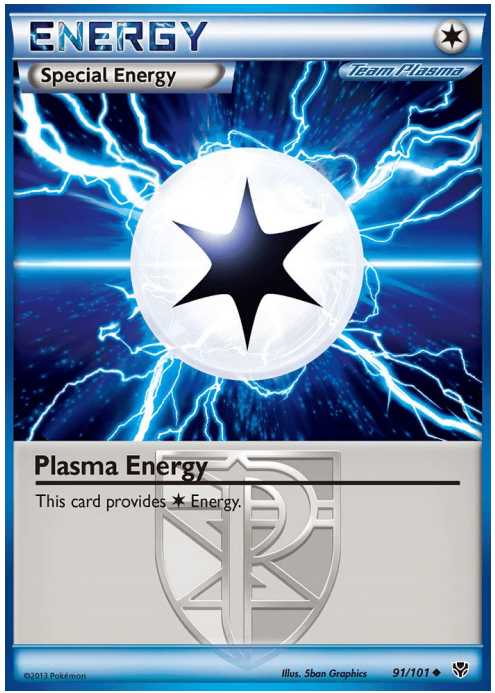


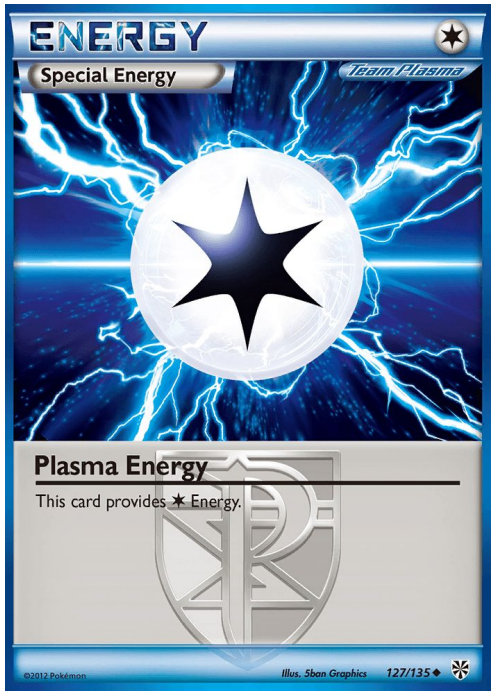


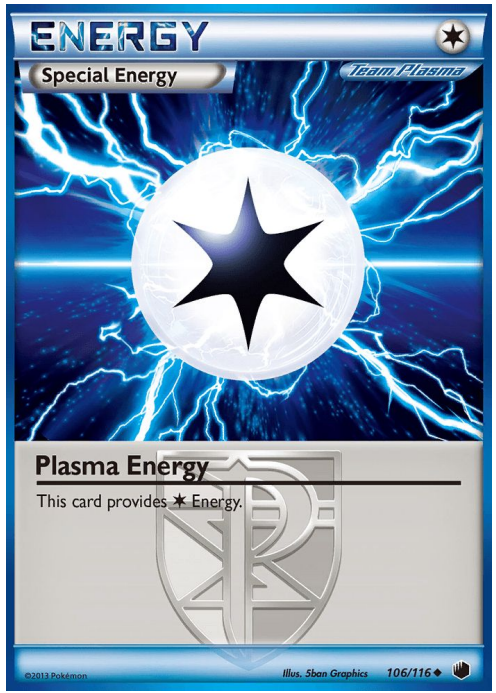




Finding a match

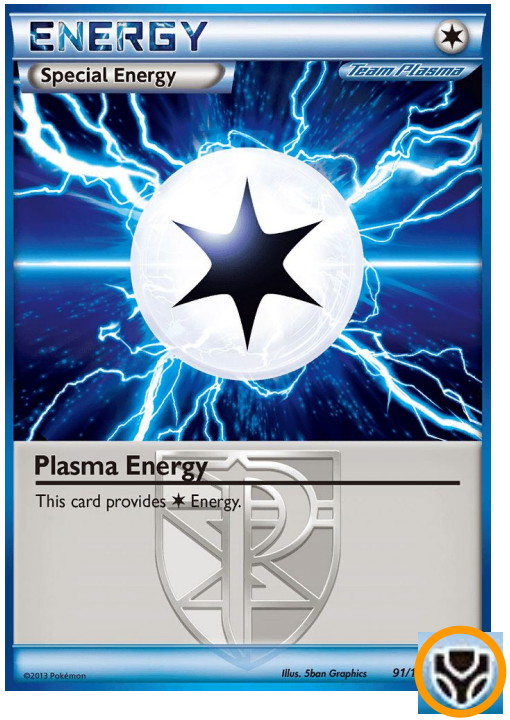


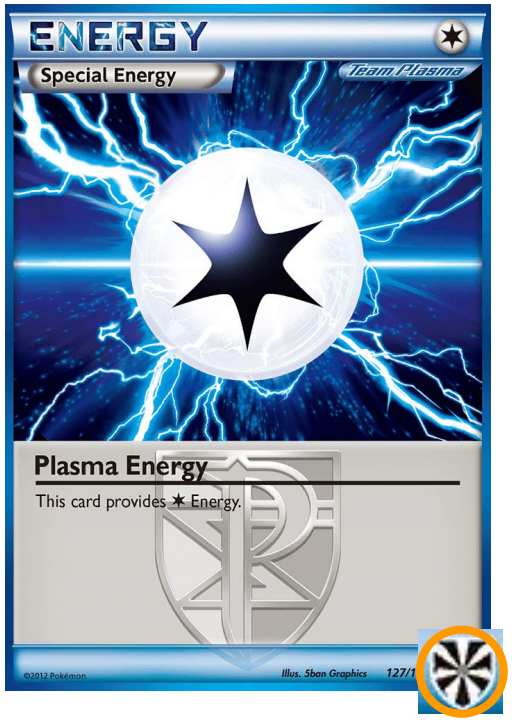




cards from different sets

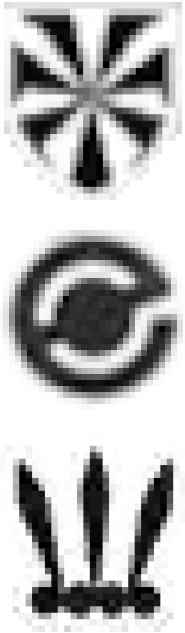
Finding a match

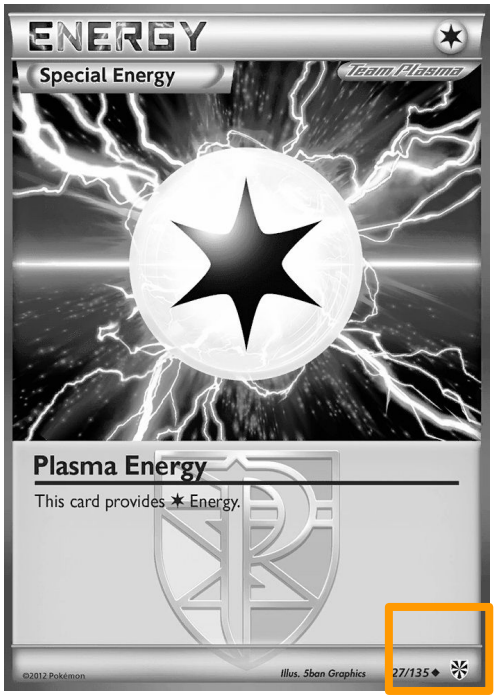




cards from different sets

Finding a match

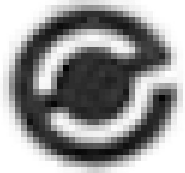




template matching

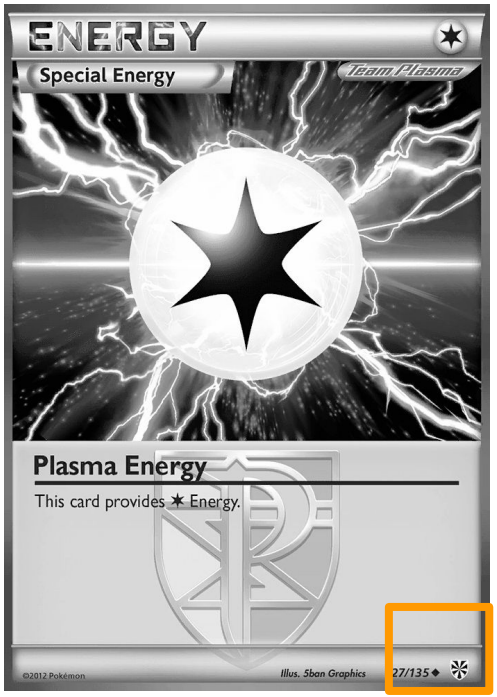
Finding a match

e < 0.15



e < 0.10

e < 0.20



template matching

Demo

Demo link: https://hugopeixoto.net/pokemon/detection/demo.mp4

Libraries

Used:

- image	https://crates.io/crates/image	image loading
- v4l	https://crates.io/crates/v4l	video stream
- nalgebra	https://crates.io/crates/nalgebra	perspective fixing
- img_hash	https://crates.io/crates/img_hash	perceptual hashing

Worth checking:

- imageproc	https://crates.io/crates/imageproc	template matching
- eye	https://crates.io/crates/eye	v4l cross platform
- rust-cv	https://github.com/rust-cv	CV community

github.com/hugopeixoto/ptcg-detection

Hugo Peixoto (he/him)

hugopeixoto.net