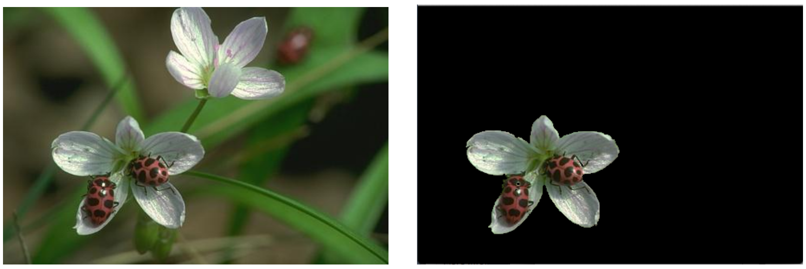
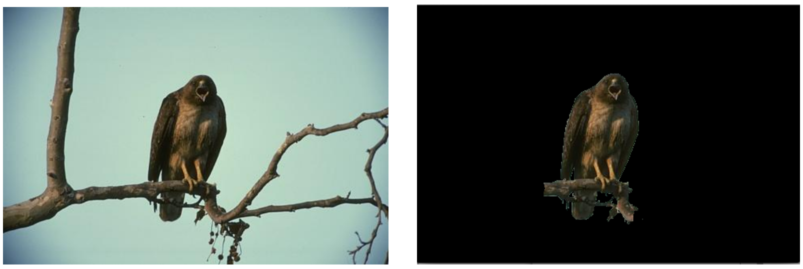


(a) The original image (b) Grab Cut

Figure 1: Grab cut on this image works very well. The petals and the bugs are extracted within 2 iterations of the algorithm. Extraction of a single bug by placing the grab rectangle around one of them is not possible (picture not shown)



(a) The original image (b) Grab Cut

Figure 2: The eagle is extracted quite nicely, although it takes up to 10-15 iterations until the sky parts between wing and body are recognized as background. As can be seen, it is also not possible to completely segment away the tree branch. This is not very surprising since the color of the bark is very similar to the color of the bird.

1 Grab Cut

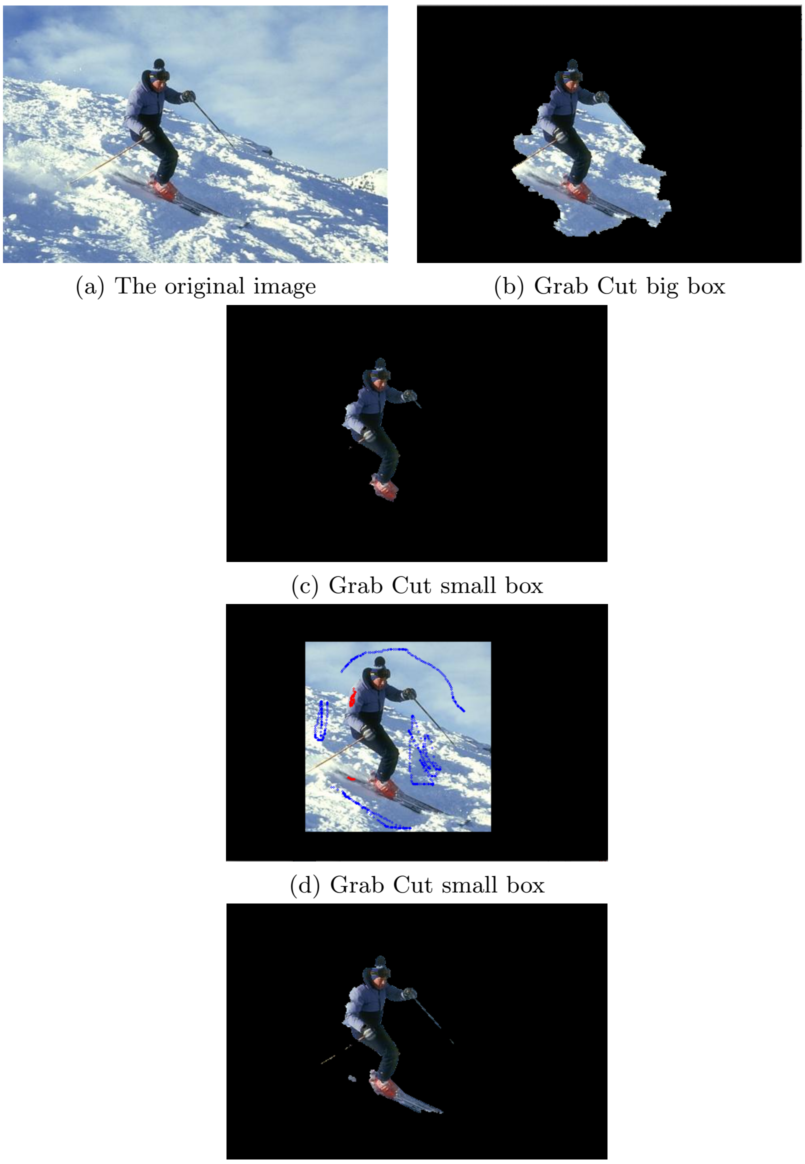
Grab cut is a very elegant algorithm for image segmentation. We were asked to use it on a series of images. Luckily, OpenCV provides us with a very neat implementation of the grab cut algorithm.

Some observations about the performance of the grab cut

- backgroun
- hello

2 Eroding and dilating

We have made the text on the Africa map bolder by eroding with a 3 by 3 rectangle as a structuring element, as seen on Fig. 12. We have then cleared



(a) The original image

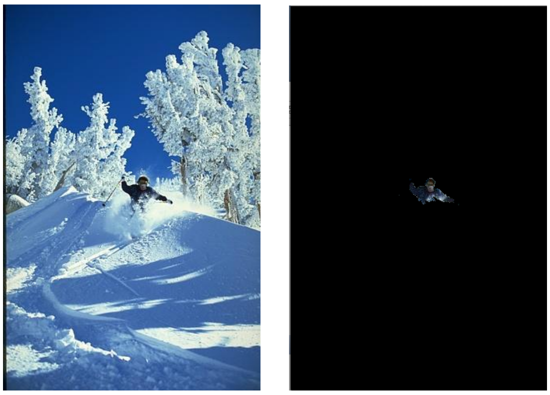
(b) Grab Cut big box

(c) Grab Cut small box

(d) Grab Cut small box

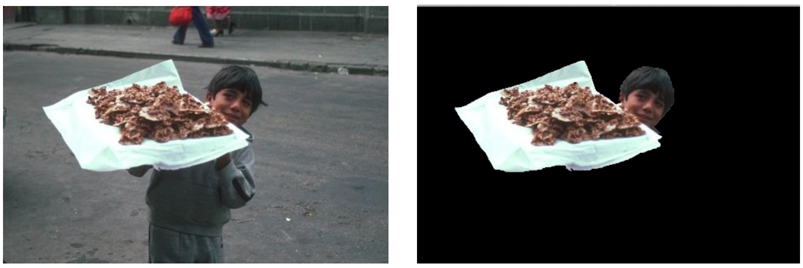
(e) Grab Cut small box

Figure 3: This image proves to be tricky to segment. When the bounding rectangle is drawin around the whole skier (including skies), the resulting segmentation stabilizes after around 10 iterations with b), which clearly is not usable. When the rectangle is only drawn around the body, it also takes around 10 iterations to result in c), which is a little better, but now lacks part of the skier. In d) the grab cut algorithm is provided with additional background and foreground pixels, the result in d) is obtained after around 5 iterations.



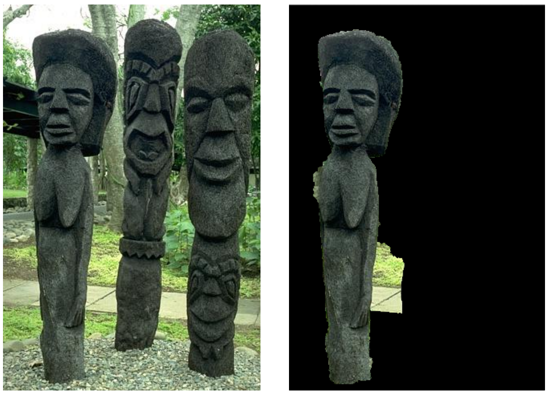
(a) The original image (b) Grab Cut

Figure 4: Clear segmentation of this image proves hard again. This is probably due to the fact that the object of interest, the skier, is partly hidden behind a cloud of snow.



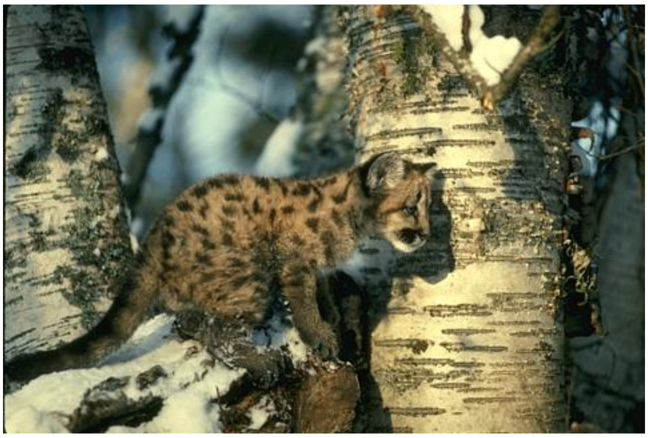
(a) The original image (b) Grab Cut

Figure 5: Clear segmentation of this image proves hard again. This is because the pants and sweather of the kid are gray, similar to the color of the street. Better results could be achieved by marking additional foreground pixels (data not shown).



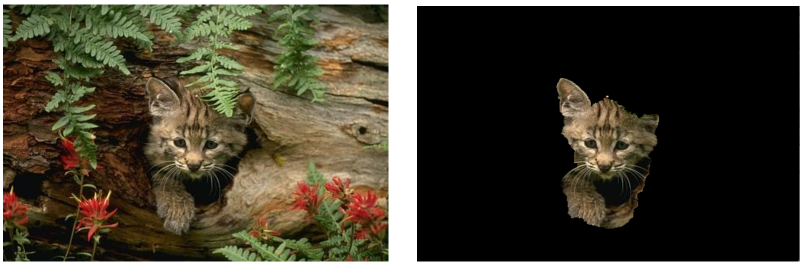
(a) The original image (b) Grab Cut

Figure 6: Clear segmentation of this image proves hard again. When all three pillars are selected, the regions in between them also get marked as foreground. If only one is selected, parts of the background always get segmented as foreground. This might be of the greeninsh-greyish color of the pillar.



(a) The original image

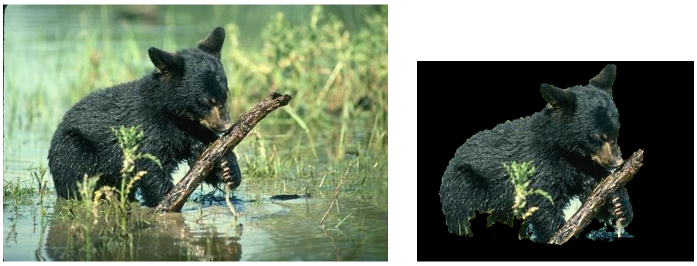
Figure 7: Segmentation of this image with the grab cut method without labeling any pixels is impossible. This is not surprising, these cats have evolved to blend into the background in order to be less visible.



(a) The original image

(b) Grab Cut

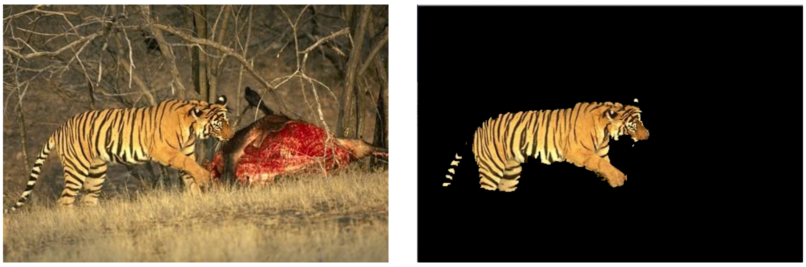
Figure 8: Clear segmentation of this image proves hard again. The kitty in the center is clearly distinguishable for us. But the fur color is similar to the bark. For the result shown in b), some additional pixels have been marked as background.



(a) The original image

(b) Grab Cut

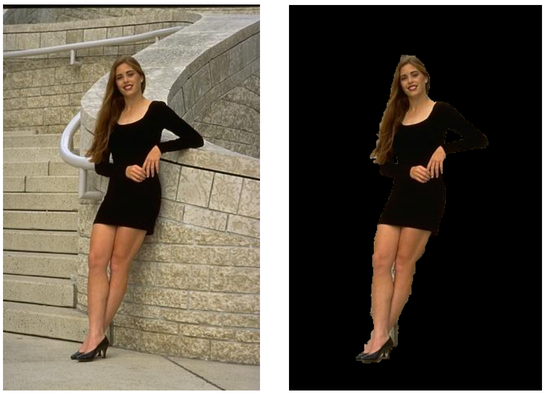
Figure 9: An acceptable segmentation of the bear is easily obtained after about 5 iterations. But the piece of grass, the log and the water seen between the legs is always included. Only seperate marking will improve segmentation (picture not shown).



(a) The original image

(b) Grab Cut

Figure 10: The segmentation of the tiger worked better as excpeted, since it also evolved to blend in. This segmentation is obtained after 4 iterations.



(a) The original image (b) Grab Cut

Figure 11: An acceptable segmentation of the woman is easy, it is a clear background/foreground relation.Also the dress is clearly segmented (not seen in this small picture).

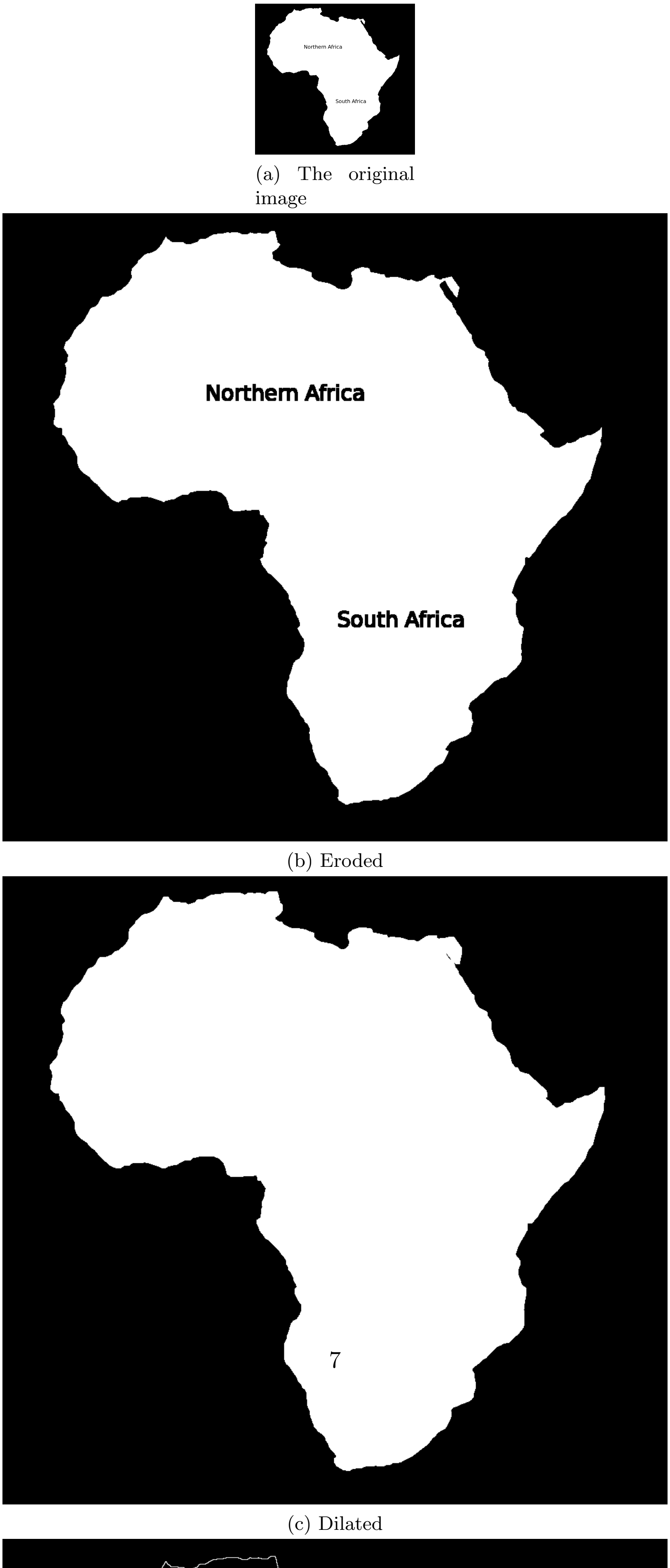
out the text by dilating the image three times with the same rectangle structuring element as before, as seen on Fig. 12. For the map outline, we cancelled out the text as before, then we eroded three times to revert the effects of dilation on the map outline, then we copied the image, eroded again and subtracted the copy from the image. The result can be seen in Fig. 12.

3 Opening and closing

By using opening with a 9 by 9 circle as a structuring element, we were able to separate the dots from the lines in Fig. 13. For the cells image, we have found that applying the same opening operation with a 7 by 7 circle after thresholding is very well suited for finding blobs above a certain size, in this case the larger cell bodies. We have also applied closing to the circles image with the same disk shaped structuring element and have discovered that this operation makes the line circles around the filled circles disappear. Making the structuring element bigger (17 by 17) will slowly get rid of the smaller dots and in the end as it gets large enough (33 by 33) it will cancel all the dots in the image.

4 Advanced

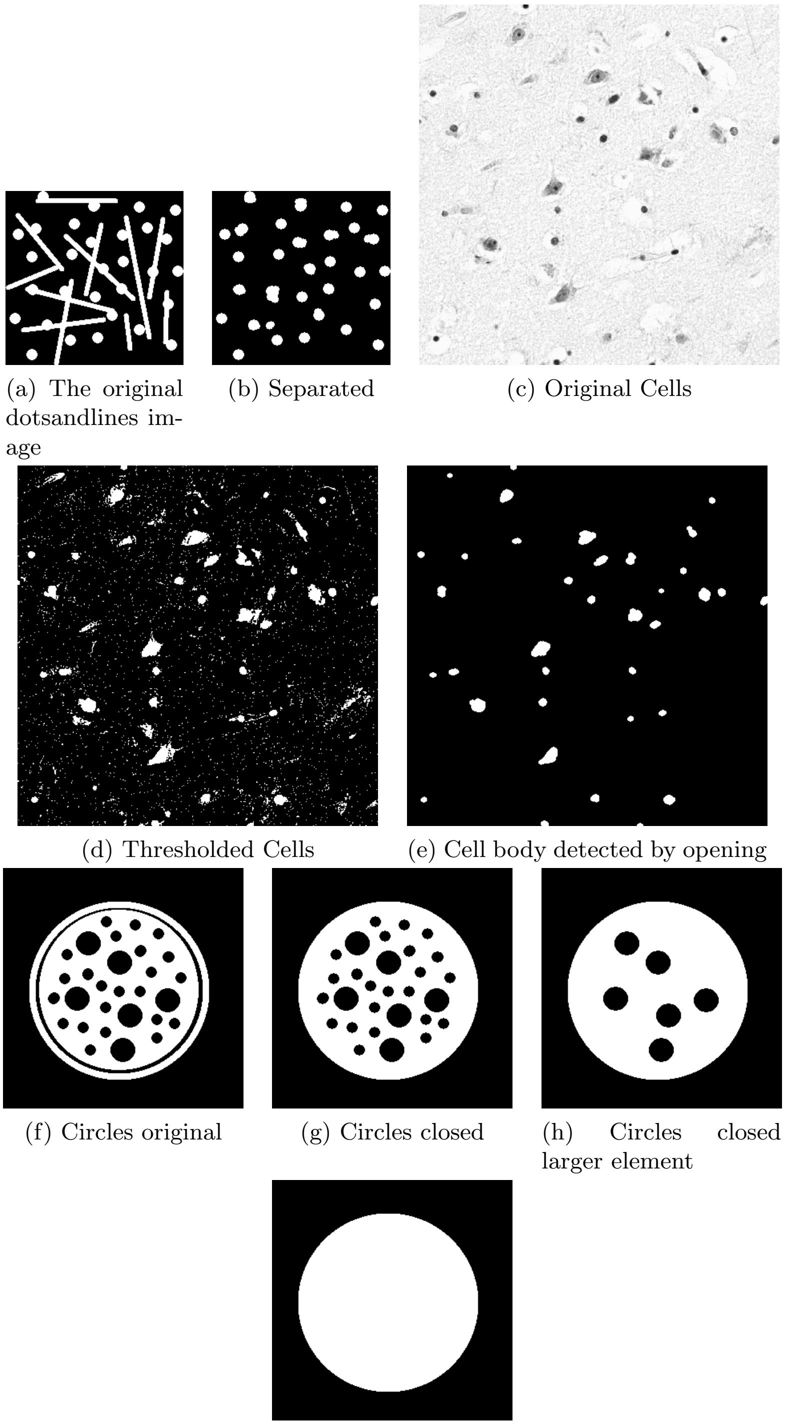
We have taken the house image (Fig. 14) and tried to implement an edge detection by using the morphological gradient with a 3 by 3 cross as structuring element. After thresholding at 60, the lines are clearly visible on the output image. Compared to Canny it performs worse, since Canny also



(a) The original image

(b) Eroded

(c) Dilated



(a) The original dotsandlines image

(b) Separated

(c) Original Cells

(d) Thresholded Cells

(e) Cell body detected by opening

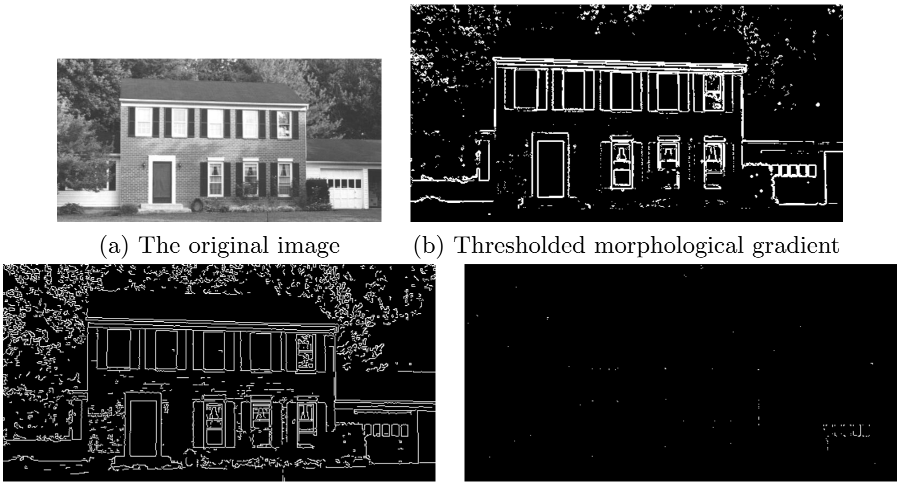
(f) Circles original

(g) Circles closed

(h) Circles closed larger element

(i) Circles closed even larger element

Figure 13: Opening and closing



(a) The original image

(b) Thresholded morphological gradient

(c) Canny

(d) Corner detection

Figure 14: Advanced morphological operations

tries to suppress non-maxima and attempts to connect different parts of a line. For corner detection, we have added the absolute value of the morphological gradient with a horizontal line as structuring element to the same with a vertical structuring element. After thresholding, regions where both gradients are strong will stay and thus we have found a corner.