

main()

Data input
Start of timer

edge_detect()

image_rgb
output

Creation of
image_gray
and **temp_buf**

rgbToGrayscale()

image_rgb
image_gray

Convert **image_rgb**
into **image_gray**

(no computations at
this stage)

convolve2d_smooth()

image_gray
output

Perform a 2D conv. of
image_gray using a
local smoothing filter,
with the result stored
in **output**

(no computations at
this stage)

convolve2d_vertical()

output
image_gray

Perform a 2D conv. of
output using local
vertical filter, with the
result stored in
image_gray

(no computations at
this stage)

convolve2d_horizontal()

output
temp_buf

Perform a 2D conv. of
output using a local
horizontal filter, with
the result stored in
temp_buf

(no computations at
this stage)

combthreshold()

image_gray
temp_buf
output

Set each position of
output to 255 or 0 if
the corresponding
positions in
image_gray and
temp_buf meet a
given threshold

(no computations in
this stage)

Array sizes:

image_rgb: int[H][W * 3]
image_gray: int[H][W]
temp_buf: int[H][W]
output: int[H][W]

Where H is the image height, w
the width, and K the filter size

An asterisk (*) next to an array
indicates the array is modified in
the function

Right-directed arrows represent
function calls, and their args.

Left-directed arrows represent
function returns

Dotted lines represent a function
waiting for a call it made to
another function to be finished.

End of timer
Write **output** to file
Verify result with
checksum