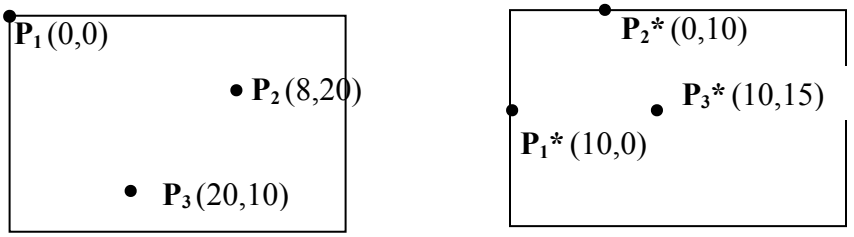
**IYTE EE 431 Intro to Image and Video Processing**
**Homework 2 Due Nov 24 2021 ŞG**

*PART A*
(This part should be submitted as a pdf file along with the c file written for part b)

**1.** Find the parameters of forward and backward affine transformations that maps points $\mathbf{P}_i$ to $\mathbf{P}_i^*$.




Hint: Write 6 equations using corresponding points and the affine model and solve for the unknowns.

**2.** Suppose that a digital image is subjected to histogram equalization. Show that a second pass of histogram equalization will produce exactly the same result as the first pass.

**3.** Develop and describe a modified Hough transformation approach to detect rectangular shapes.

*PART B COMPUTER PROBLEM*

Develop a computer program that performs median filtering on an image. The size of the filter mask and number of iterations should be entered in the command line. To follow the same structure as the example programs you may need to create a file that contains the function (e.g. `median.c`) and an application program (e.g. `median-app.c`) which can be modified from "`example*.c`". You will also need to update the header file "`my_header.h`" to include the function declarations. As described in the README file the compilation should be done by:

```
> gcc -o median-app median-app.c median.c img_pro.c
```

If you need to enable debugging you will need the flag (-g):

```
> gcc -g -o median-app median-app.c median.c img_pro.c
```

In unix systems you will need to use (-lm) at the end of the line to link with the math library if necessary.

Your application program should accept 3 arguments in the command line:

```
> ./median-app 7 2 [.pgm file]
```

means the size of the mask is 7x7 and the filter should be applied twice. The application program should be able to handle filters of size 3x3, 5x5 and 7x7.

Sample images that can be used for this homework can be found in http://www.fit.vutbr.cz/~vasicek/imagedb/

The solutions to computer problems should be submitted by the previously assigned groups via course Teams site at or before the due date/time.

**Note**: You may come across some run time errors (e.g. "segmentation faults") while working on your programs. To debug your code compile your code with the debug flag set and run your program under the debugger program:

```
> ./gdb median-app
(gdb) r 7 2 [.pgm file]
```

This should point the faulty lines of your code that cause the crash. At the level of the crash you can see the values of the variables by using "print [varname]".