

❖ Develop a program that has the following three functions.

- ✓ Input: 'Faces.mp4'
 - It is up to you whether you read it as grayscale or not. (but colorful video recommended)
- ✓ Output: A video window showing the result according to each function
 - Repeat the video playback.
- ✓ All functions need to be implemented in one single program.
- ✓ How you implement this program is totally up to you, which means you can use any methods as long as the results satisfy the function requirements.
- ✓ Record a working video showing all the results according to the functions, upload the video on YouTube, and write down the video address in your report.

HW9

❖ Develop a program that has following functions.

✓ Function 1 (7 points)

- When user presses 'b' key, subtract the background from the original image (so that the only human regions will remain).
- When user presses 'b' key again, return to the original video.
- During background subtraction, count the number of people and write the number on the result window.
 - » Use morphological operation (for this case, do not use face detection)
 - » You may use tracking techniques.



Note that this is just an example.
Your result may be different.

HW9

❖ Develop a program that has following functions.

✓ Function 2 (5 points)

- When user presses 'f' key, detect all faces and draw a rectangle box on every face region.
 - » Use green color for the nearest face, blue color for the farthest face, and yellow color for the other face.
- When user presses 'f' key again, remove all boxes.



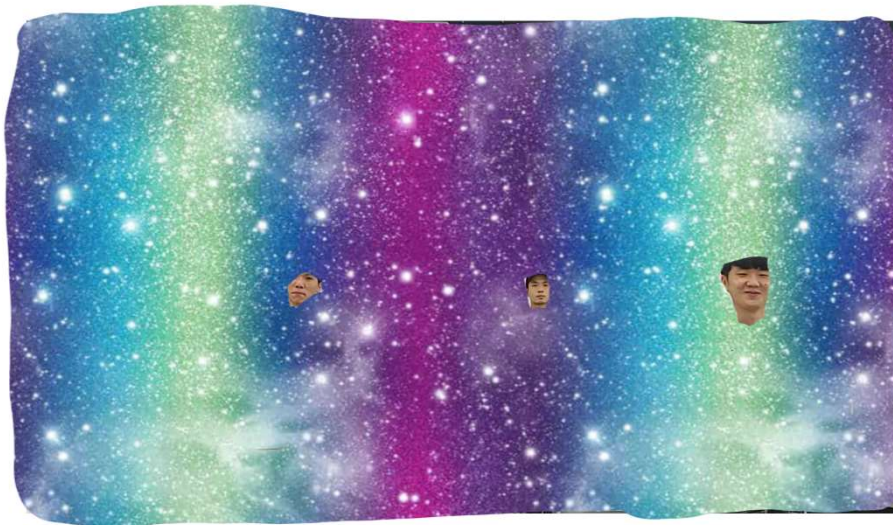
Note that this is just an example.
Your result may be different.

HW9

❖ Develop a program that has following functions.

✓ Function 3 (8 points)

- When user presses 'g' key, detect all faces areas (not rectangular area) and fill the rest with a virtual background.
 - » The virtual background image can be arbitrarily.
 - » You may use grabCut() function for segmenting a face area. (refer to the next slide)
- When user presses 'g' key again, return to the original video.



Note that this is just an example.
Your result may be different.



HW9

❖ Example code

```
Mat result, bgModel, fgModel, image, foreground;

image = imread("assets/dog.png");
//inner rectangle which includes foreground
Rectrectangle(65, 0, 170, 230);
grabCut(image, result, rectangle, bgModel, fgModel, 10, GC_INIT_WITH_RECT);
compare(result, GC_PR_FGD, result, CMP_EQ);

foreground = Mat(image.size(), CV_8UC3, Scalar(255, 255, 255));

image.copyTo(foreground, result);
imshow("original", image);
imshow("Result", result);
imshow("Foreground", foreground);
```

