

SELFIE-AID APP

Project Report – Selfie Aid App

Catalogue :

1. *Project introduction*

2. *Objectives*

- *Easy to use*
- *Delight to use*
- *Fun to use*

3. *Instructions (Graphical User Interface & Usage Directions)*

- *Main GUI*
- *Snapshot GUI*
- *Video GUI*

3. *Functions and Demos*

- *Snapshot*
 - *Filters*
 - *Blur*
 - *Brighten*
 - *Drawing*
 - *Blue rectangle*
 - *Blue circle*
 - *Any color of circle*
 - *Stickers*
 - *Random color lines*
 - *Rainbow lines*
 - *Hat stickers*
 - *Face and eyes recognition*
 - *Face and Eyes recognition*
 - *Add sunglasses*
 - *Add cartoon glasses*
 - *Cartoon figures*
 - *Mickey Mouse*
 - *Nick*
 - *Tigger*
 - *Return*
 - *Return to the original photo*

SELFIE-AID APP

-
- *Save and Quit*
 - *Save and Quit*
 - *Video*
 - *Filters*
 - *Brighten*
 - *Cancel Brighten*
 - *Recording*
 - *Start Recording*
 - *Pause Recording*
 - *Effects*
 - *Nick and Judy*
 - *Mickey and Minnie*
 - *Tigger and Eeyore*
 - *Mustache*
 - *Save and Quit*
 - *Save and Quit*

4. Problems and Solutions

- *GUI*
 - *Sub Windows*
 - *Color*
- *Snapshot*
 - *Brightness*
 - *Colorful line*
 - *Face and eyes recognition*
 - *Transparent stickers*
 - *Resize the stickers*
- *Video*
 - *Flags*
 - *Improvements after presentation*

5. Conclusion

SELFIE-AID APP

Project introduction :

Nowadays, we have so many fun selfie APPs such as Faceu, Snapchat and Instagram, people enjoys creating their selfies with their own style.

Our group is formed with two girls, we were inspired these delightful APPs. We aimed to design our own Selfie-Aid APP that can provide fun effects either on photos or videos. Also, we noticed the importance of facial beautification and used suitable filter to achieve this automatically.

Objectives:

Easy to use:

All the outputs are named as the time they are created, it's simple and easy to find, so users won't confused with our naming system.

We have usage directions shown while users using them, they just need to follow our instructions, no need to find or memorize any of the instructions.

Delightful to use:

We put much effort to making our GUI colorful and nice. Every single instruction has its own color, and all colorful lines form as a harmonious entirety.

Fun to use:

Our APP has two main branches: Photos and Short Videos.

For Photos:

All the photos will be stored after both snapshotting and image-processing. Users can draw creatively with our provided functions, they can also choose cute stickers we offered. We have 'return' function as well, they can always return to the original photo if they don't like the affects.

For Short Videos:

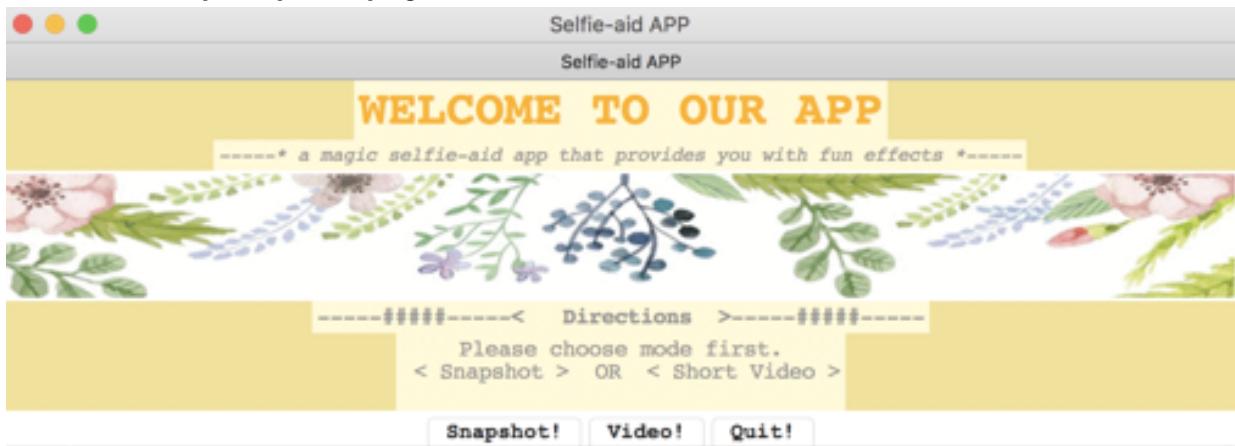
We provide video recording and pause functions, after start recording, users can pause recording and then continue recording anytime they want. The main function in the Video part is real-time face-recognition, we have several cartoon characters for users to choose, such as Nick and Judy from Zootopia, they can show as these cute characters in the video.

SELFIE-AID APP**Instructions (Graphical User Interface & Usage Directions)**

Main GUI:

Main GUI is the first GUI after running the program. It demonstrate what our app use for and direct the user to choose the mode.

- Snapshot : Go to the snapshot function
- Video : Go to the video function
- Quit : Stop and quit this program

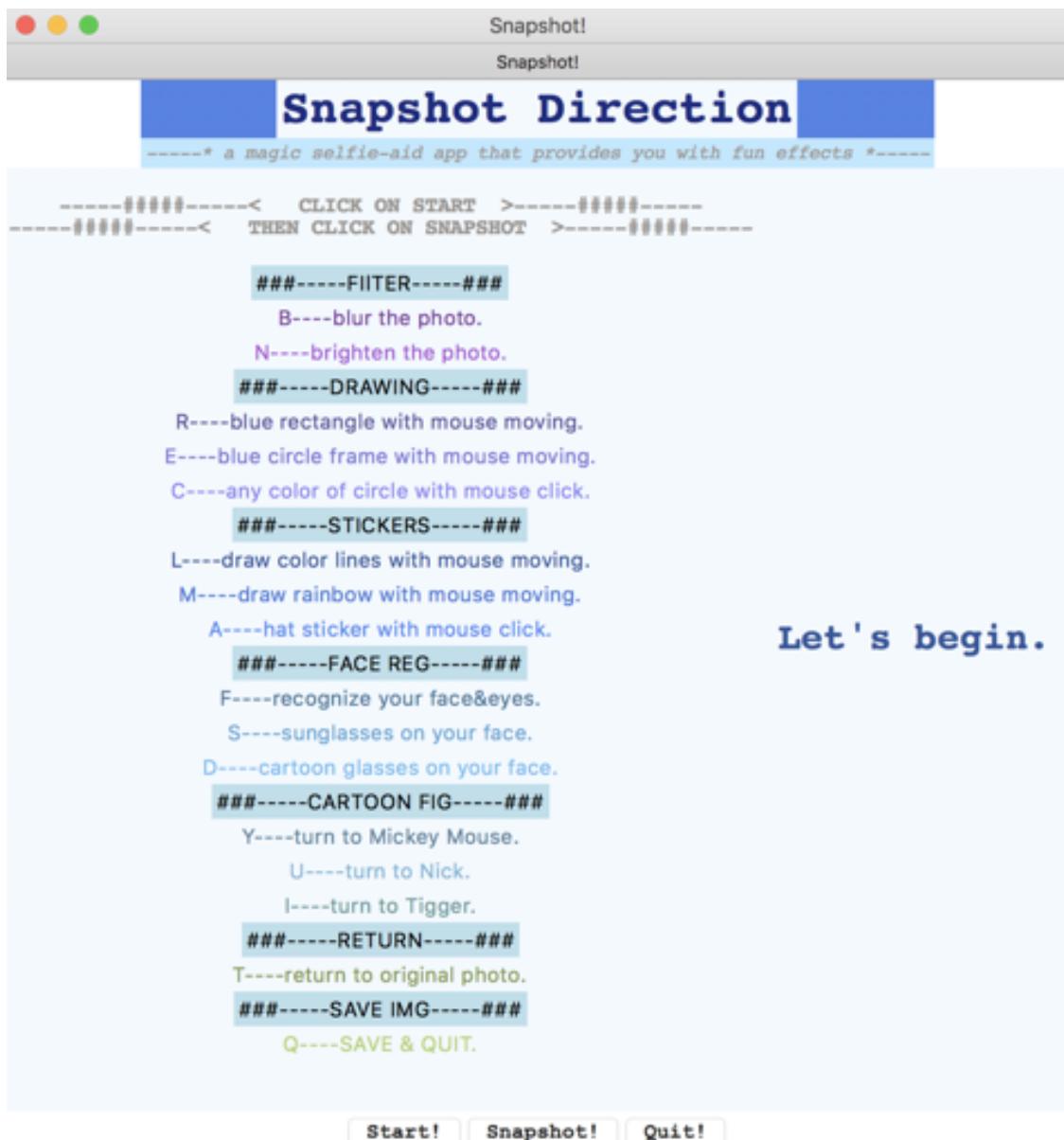


SELFIE-AID APP***Snapshot GUI:***

Snapshot GUI appears after click on the snapshot button. It demonstrates what can we do in the Snapshot mode.

If users click on the start button, the program will call the camera function in the computer and show the image on the GUI in real-time. Users are encouraged to click snapshot button at any time they looks nice in the image.

After snapshot, an CV2 window will pop up with the photo that the user had taken. Our project provides many different image processing functions in this section. Users can use the keyboard to select different functions and operate with mouse moving or mouse click.

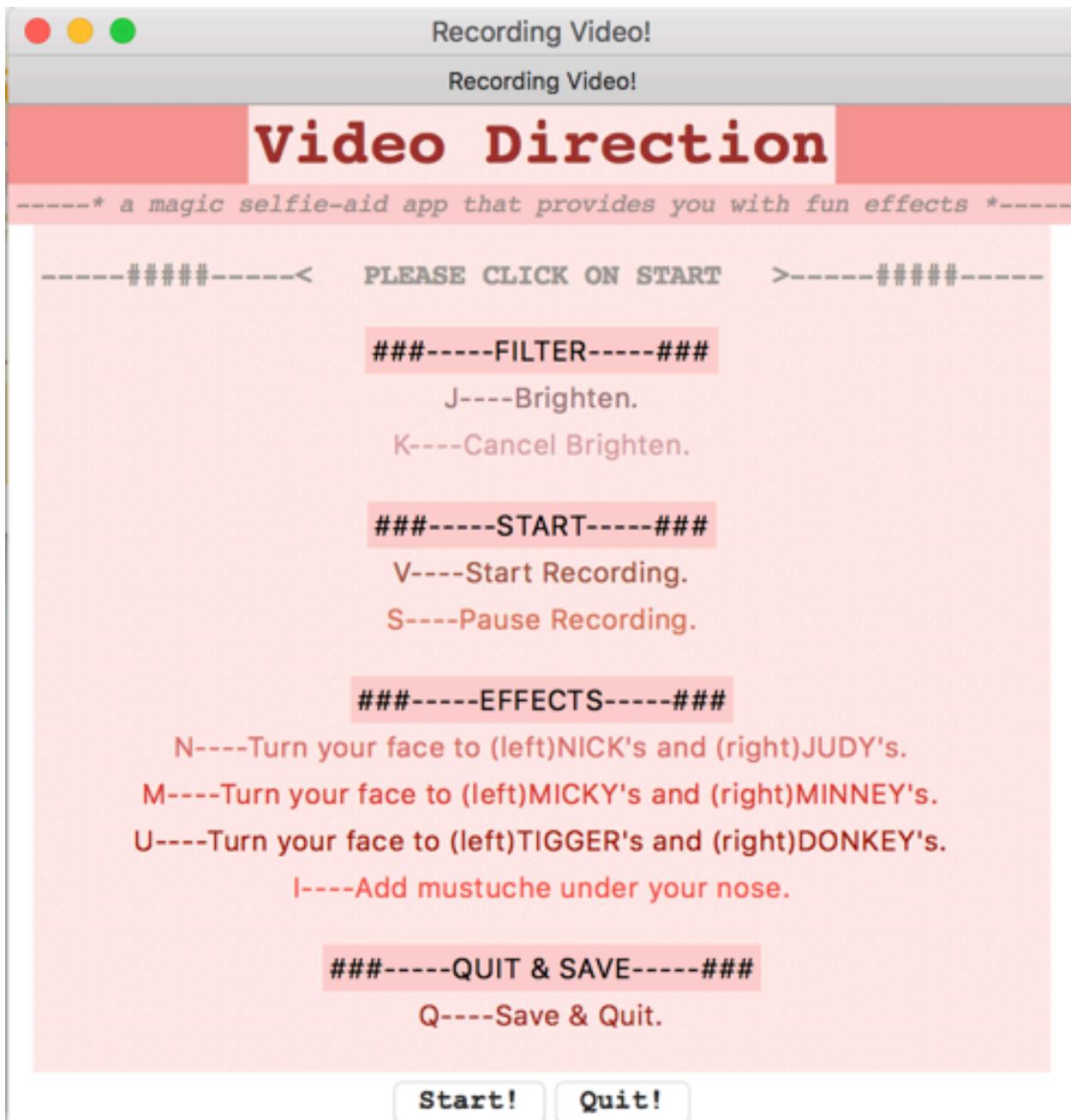


SELFIE-AID APP

Video GUI :

Video GUI appears after click on the video button. It demonstrates what can we do in the Video mode.

If users click on the start button, the program will call the camera function in the computer and an CV2 window will pop up with real-time video.



SELFIE-AID APP

Functions and Demos

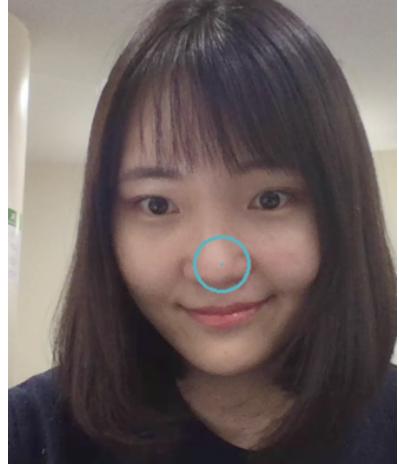
Snapshot:

1. Filters

	Function	Demo
Blur	<p><i>This function allows the user to blur the photo with</i></p> $\text{kernel} = \text{np.ones}(3,3), \text{np.float32})/9$	
Brighten	<p><i>This function allows the user to brighten the photo with</i></p> $\text{gamma} = 1.2..$ <p><i>When gamma > 1, the photo will brighter.</i></p> <p><i>When gamma < 1, the photo will darker.</i></p> <p><i>Formula:</i></p> $\text{invGamma} = 1.0 / \text{gamma}$ $\text{table} = \text{np.array}([(i / 255.0) ** \text{invGamma}) * 255 for i in np.arange(0, 256)]).astype("uint8")$ $\text{return cv2.LUT(img, table)}$	

SELFIE-AID APP

2. Drawing

	Function	Demo
Blue Rectangle	<p><i>This function allow users to draw a light green blue rectangle.</i></p> <p><i>This function use cv2.rectangle.</i></p>	
Blue Circle frame	<p><i>This function allow users to draw a light blue circle frame.</i></p> <p><i>This function use cv2.circle.</i></p>	
Any Color of Circle	<p><i>This function allow users to draw a circle with any color. Users can use the Trackbar to control the value of each BGR channel.</i></p> <p><i>This function use cv2.getTrackbarPos.</i></p>	

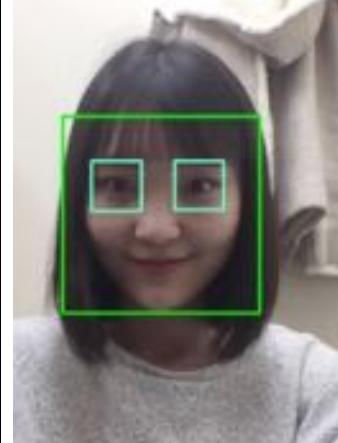
SELFIE-AID APP

3. Stickers

	<i>Function</i>	<i>Demo</i>
<i>Color lines</i>	<i>This function allows users to draw color lines with mouse moving, the color of the line is changed automatically according to the position of (x,y)</i>	
<i>Rainbow</i>	<i>This function allows users to draw sticker lines with mouse moving, the rainbow was made by stickers.</i>	
<i>Hat Stickers</i>	<i>This function allows users to add hat stickers by mouse clicking.</i>	

SELFIE-AID APP

4. Face and eyes recognition

	Function	Demo
Face and eyes recognition	<p><i>This function allows users to check if their face and eyes are recognized by the import library.</i></p> <p><i>This function uses cv2.CascadeClassifier.</i></p>	
Add sunglasses	<p><i>This function allows users to add sunglasses on their face.</i></p> <p><i>The glasses can be resized automatically according to the face shape.</i></p>	
Add cartoon glasses	<p><i>This function allows users to add cartoon glasses on their face.</i></p> <p><i>This cartoon glasses can be resized automatically according to the face shape.</i></p>	

SELFIE-AID APP

5. Cartoon figures

	Function	Demo
Mickey Mouse	<p><i>This function allows users to turn to cartoon character -- Mickey Mouse.</i></p> <p><i>This cartoon figure can be resized automatically according to the face shape.</i></p>	
Nick	<p><i>This function allows users to turn to cartoon character -- Nick.</i></p> <p><i>This cartoon figure can be resized automatically according to the face shape.</i></p>	
Tigger	<p><i>This function allows users to turn to cartoon character -- Tigger.</i></p> <p><i>This cartoon figure can be resized automatically according to the face shape.</i></p>	

SELFIE-AID APP

6. Return

Return function allowed the user to return to the original photo.

7. Save and quit

Save and quit function allowed the user to save the processed image and quit the Snapshot mode.

SELFIE-AID APP

Video:

1. Filter

	Function	Demo
Brighten	<i>This function allows users to brighten the video.</i>	
Cancel brighten	<i>This function allows users to cancel brighten the video.</i>	

2. Recording

Record videos whenever pressing 'v' and can always pause recording by pressing 's', press 'v' again to go on recording.

SELFIE-AID APP

3. Effects

	Function	Demo
<i>Nick and Judy</i>	<p><i>This function allows users to turn to cartoon character -- Nick(left side) & Judy(right side).</i></p> <p><i>This cartoon figures can be resized automatically according to the face shape.</i></p>	
<i>Mickey and Minnie</i>	<p><i>This function allows users to turn to cartoon character -- Mickey(left side) & Minnie(right side).</i></p> <p><i>This cartoon figures can be resized automatically according to the face shape.</i></p>	
<i>Tigger and Eeyore</i>	<p><i>This function allows users to turn to cartoon character -- Tigger(left side) & Eeyore(right side).</i></p> <p><i>This cartoon figures can be resized automatically according to the face shape.</i></p>	
<i>Mustache</i>	<p><i>This function allows users to add cartoon mustache under the nose.</i></p> <p><i>This cartoon figures can be resized automatically according to the face shape.</i></p>	

4. Save and quit

Save and quit function allowed the user to save the processed video and quit the Video mode.

SELFIE-AID APP

Problems and Solutions

1. GUI:

a. Sub windows

We use TKinter for our GUI, the first problem is that we need TKinter to generate new sub windows for our different function mode.

This is solved by create different 'top' using Tk.Toplevel() other than Tk.Tk().

b. Colors

I was really unsatisfied with the color like red, blue, green and etc. It looks kind of strange and it is difficult to mix them together.

I searched on the website and got the following color table.

Named colour chart													
snow	deep sky blue	gold	seashell3	SteelBlue2	LightBlue3	SpringGreen2	DarkGoldenrod1	brown4	pink3	purple1	gray8	gray64	
ghost white	sky blue	light goldenrod	seashell4	SteelBlue3	LightBlue4	SpringGreen3	DarkGoldenrod2	salmon1	pink4	purple2	gray6	gray65	
white smoke	light sky blue	goldenrod	AntiqueWhite1	SteelBlue4	LightCyan2	SpringGreen4	DarkGoldenrod3	salmon2	LightPink1	purple3	gray12	gray66	
gainsboro	steel blue	dark goldenrod	AntiqueWhite2	RoyalBlue1	LightCyan3	green2	DarkGoldenrod4	salmon3	LightPink2	purple4	gray29	gray67	
Floral white	light steel blue	rose brown	AntiqueWhite3	RoyalBlue2	LightCyan4	green3	DarkGoldenrod5	RosyBrown1	salmon4	LightPink3	MediumPurple1	gray30	gray68
old lace	light blue	indian red	AntiqueWhite4	RoyalBlue3	PaleTurquoise1	green4	DarkGoldenrod6	RosyBrown2	LightSalmon2	LightPink4	MediumPurple2	gray31	gray69
linen	powder blue	saddle brown	Ivory2	RoyalBlue4	PaleTurquoise2	chocolate2	DarkGoldenrod7	RosyBrown3	LightSalmon3	LightPink5	MediumPurple3	gray32	gray70
antique white	pale turquoise	sandy brown	Ivory3	RoyalBlue5	PaleTurquoise3	chocolate3	DarkGoldenrod8	RosyBrown4	LightSalmon4	LightPink6	MediumPurple4	gray33	gray71
papaya whip	dark turquoise	dark salmon	Ivory4	RoyalBlue6	PaleTurquoise4	chocolate4	DarkGoldenrod9	IndigoRed1	orange2	LightPink7	MediumPurple5	gray34	gray72
Blanched almond	medium turquoise	salmon	PeachPUt2	DodgerBlue2	CadetBlue1	OliveDrab2	IndigoRed2	orange3	LightPink8	MediumPurple6	MediumPurple6	gray35	gray73
bisque	turquoise	light salmon	PeachPUt3	DodgerBlue3	CadetBlue2	OliveDrab3	IndigoRed3	orange4	maroon1	orange5	gray36	gray74	
peach puff	cyan	orange	PeachPUt4	DodgerBlue4	CadetBlue3	OliveDrab4	IndigoRed4	DarkOrange2	maroon2	orange6	gray37	gray75	
navejo white	light cyan	dark orange	NavyBlue2	SteelBlue1	Turquoise1	DarkOliveGreen1	sienna1	DarkOrange3	maroon3	orange7	gray38	gray76	
cadet chiffon	cader blue	coral	NavyBlue3	SteelBlue2	Turquoise2	DarkOliveGreen2	sienna2	DarkOrange4	maroon4	orange8	gray39	gray77	
mint cream	medium aquamarine	light coral	NavyBlue4	SteelBlue3	Turquoise3	DarkOliveGreen3	sienna3	DarkOrange5	ViolentRed1	orange9	gray40	gray78	
azure	aquamarine	tomato	LemonChiffon2	SteelBlue4	Turquoise4	DarkOliveGreen4	sienna4	DarkOrange6	ViolentRed2	orange10	gray41	gray79	
alice blue	dark green	orange red	LemonChiffon3	DeepSkyBlue2	Turquoise5	khaki1	burlywood1	coral1	ViolentRed3	orange11	gray42	gray80	
lavender	dark olive green	red	LemonChiffon4	DeepSkyBlue3	Turquoise6	khaki2	burlywood2	coral2	ViolentRed4	orange12	gray43	gray81	
lavender blush	dark sea green	hot pink	LemonChiffon5	DeepSkyBlue4	Turquoise7	khaki3	burlywood3	coral3	ViolentRed5	orange13	gray44	gray82	
emily rose	sea green	deep pink	LemonChiffon6	DeepSkyBlue5	Turquoise8	khaki4	burlywood4	coral4	magenta2	orange14	gray45	gray83	
dark slate gray	medium sea green	pink	LemonChiffon7	SkyBlue2	DarkSlateGray1	LightCoral1	wheat1	coral5	magenta3	orange15	gray46	gray84	
dim gray	light sea green	light pink	LemonChiffon8	SkyBlue3	DarkSlateGray2	LightCoral2	wheat2	coral6	orchid1	orange16	gray47	gray85	
state gray	pale green	pale violet red	LemonChiffon9	SkyBlue4	DarkSlateGray3	LightCoral3	wheat3	coral7	orchid2	orange17	gray48	gray86	
light slate gray	spring green	maroon	LemonChiffon10	LightSkyBlue1	DarkSlateGray4	LightCoral4	wheat4	coral8	orchid3	orange18	gray49	gray87	
gray	teal green	medium violet red	Honeydew2	LightSkyBlue2	AquaMarine2	LightCoral5	tan1	coral9	orchid4	orange19	gray50	gray88	
light gray	medium spring green	violet red	Honeydew3	LightSkyBlue3	AquaMarine3	LightCoral6	tan2	red2	plum1	orange20	gray51	gray89	
light gray	green yellow	medium orchid	Honeydew4	LightSkyBlue4	DarkSeaGreen1	LightCoral7	tan3	red3	plum2	orange21	gray52	gray90	
light gray	lime green	dark orchid	LavenderBlush2	SteelGrey1	DarkSeaGreen2	yellow2	chocolate1	red4	plum3	orange22	gray53	gray91	
light gray	yellow green	dark violet	LavenderBlush3	SteelGrey2	DarkSeaGreen3	yellow3	chocolate2	DeepPink2	plum4	orange23	gray54	gray92	
light gray	forest green	blue violet	LavenderBlush4	SteelGrey3	DarkSeaGreen4	yellow4	chocolate3	DeepPink3	MediumOrchid1	orange24	gray55	gray93	
slate blue	olive drab	purple	MistyRose2	SteelGrey4	DarkSeaGreen5	yellow5	chocolate4	DeepPink4	MediumOrchid2	orange25	gray56	gray94	
medium slate blue	dark khaki	medium purple	MistyRose3	LightSteelBlue1	SeaGreen1	gold1	firebrick1	DeepPink5	MediumOrchid3	orange26	gray57	gray95	
light slate blue	khaki	khaki	MistyRose4	LightSteelBlue2	SeaGreen2	gold2	firebrick2	HotPink1	MediumOrchid4	orange27	gray58	gray96	
medium blue	pale goldenrod	azur2	MistyRose5	LightSteelBlue3	SeaGreen3	gold3	firebrick3	HotPink2	MediumOrchid5	orange28	gray59	gray97	
navy blue	light goldenrod yellow	azur3	MistyRose6	LightSteelBlue4	SeaGreen4	gold4	firebrick4	HotPink3	HotPink1	DarkOrchid1	gray60	gray98	
blue	light yellow	azur4	MistyRose7	LightSteelBlue5	SeaGreen5	gold5	firebrick5	HotPink4	DarkOrchid2	DarkOrchid2	gray61	gray99	
steel blue	yellow	seashell2	SteelBlue1	LightSteelBlue6	SeaGreen6	gold6	firebrick6	HotPink5	DarkOrchid3	DarkOrchid3	gray62	gray100	

2. Snapshot

a. Brightness

In order to show the image from cv2 to GUI, we converted colors from BGR to RGBA, this command made the picture seem bright in GUI but dark in the cv2 window that pop up after snapshot. I defined two brightness functions:

`def beautybrightness(self,img, gamma) --- Used after snapshot, gamma was fixed to 1.4. This function was automatically applied to the image when show the photo in cv2.`

`def brightness(self,img, gamma) --- Used in Brightening function, gamma was fixed to 1.2. This function was applied when keyboard 'N' was pressed, this operation can do more than once.`

Both of these functions were based on OpenCV LUT function. It is a 1 dimensional LookUpTable function, we can map the LUT for an 1 dimensional array consisting of a Single-channel or Multi-channel element. The core parameter is gamma, it used as the alpha layer. (RGBA, A- alpha)

Code:

```
invGamma = 1.0 / gamma
table = np.array([(i / 255.0) ** invGamma * 255 for i in np.arange(0, 256)]).astype("uint8")
return cv2.LUT(img, table)
```

b. Colorful line

We want users to draw lines not in a single color, so we make a colorful line function, all the colors are changing with the mouse position.

c. Face and eyes recognition

We use the library CascadeClassifier, it's a library in CV2 that can detect people's faces based on large data set. first change image to grayscale and then use faceCascade.detectMultiScale() to detect faces. All the stickers on face is rescaled and placed based on the face size, we will talk about details later.

d. Transparent stickers

CV2 can only support images without transparent layer, which means we can only add stickers in rectangle shape, but all our stickers have irregular shape. What we did is to generate a mask, we first cover the transparent part of stickers with a mask, then use mask to get all parts from the original photo which should not be covered by stickers(will be covered by mask if adding rectangle stickers), we combine the stickers without mask and the part of original photo that not supposed to be covered, after the combination, we can finally get the effect we want.

e. Resize the stickers

We add the cartoon figures to the photo based on the face recognition, but sometimes the cartoon figures are not the appropriate size.

Our group resize the cartoon figure by specific 1.X times bigger than the height and width of faces in the photo. Then calculate the coordinate to make the cartoon figure exactly suitable for the face in the photo. Each cartoon figure has its own resize multiplier and its own resize method. Sometimes we use the same multiplier on both height and width of the cartoon figure, sometimes we use the different multiplier for it. All of these was set to avoid much distortion of the origin cartoon figure.

If the cartoon figure exceed the frame of the photo, we will set the value to the limit value.

More details are shown in the program.

3. Video:

a. Flags

We used several flags to separate different functions, for example, if users turned on the 'Nick' function ('Nick' flag on), when they changed to 'Tiger', the 'Nick' face will gone, this is done by setting 'Nick' flag off.

b. Improvements after presentation

To make more fun, we changed the functions after presentation, when users' faces on the right side of window, our effect will be different from the effect when their faces on the left side. This can also show difference when two friends using the APP and their faces locating different side of the window, they can have different effect on their faces.

Conclusion

We had so much fun when we trying to complete this project, there were many hard times, especially when we want to add a video mode functions and when we find out CV2 doesn't support transparent layer of images, we spent so much time and efforts on solving these problems, and learnt a lot.

Basically, we did achieve all the goals we set before starting doing this project, and we had more thoughts and functions than before, this selfie-aid APP is completed and we are proud of it.