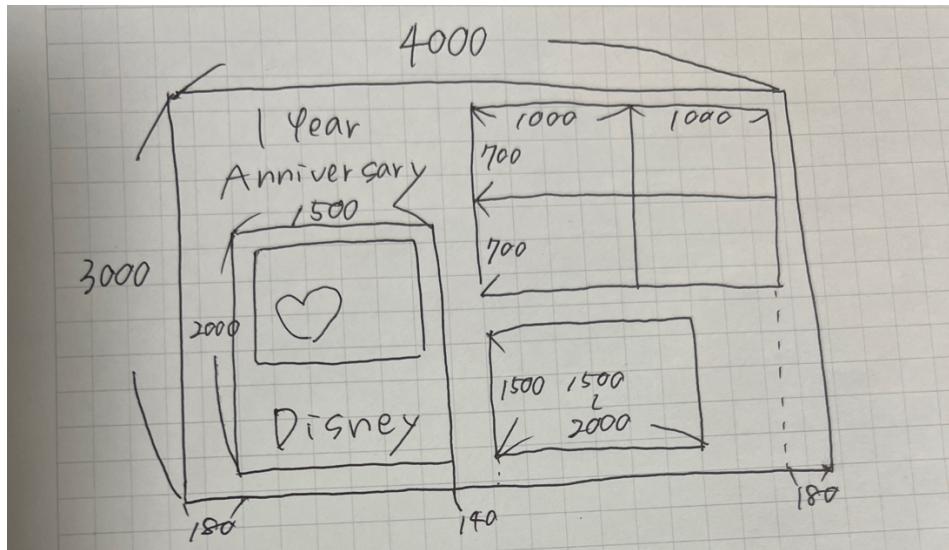


コンピュータビジョン課題 説明書

1. 完成画像



2. 計画図



3. コンセプト

『カップルフォトアルバム作成（捏造）』

クリスマスが近づく中、彼氏がない状況を少しでも楽しむ方法として、イマジナリー彼氏との幸せな時間を疑似体験できるカップルフォトアルバムを作成することにした。この試みの結果、満足のいくアルバムを作ることができたが、どこか虚しさも心に残った。それでも、もし未来の彼氏と1周年を迎えることができたときには、「本物のアルバム」を作りたいと願っている。その際には、この経験や技術が役立つことだろう。

3. 使用した元画像



mickey.jpg

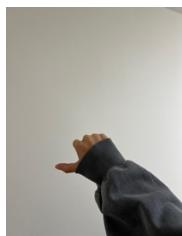


miney.jpg

(<https://store.disney.co.jp/fashion/goods/headband/>)



disney.jpg



right.jpg



left.jpg



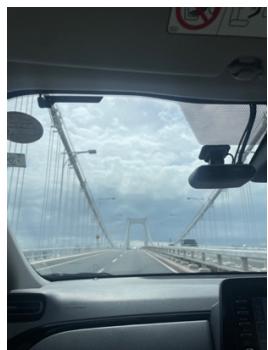
man.jpg(父の靴)



lady.jpg(自分の靴)



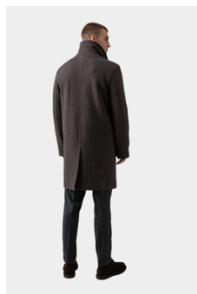
icho.jpg



car.jpg

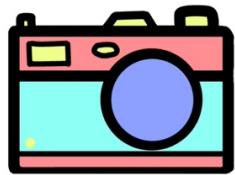


darts.jpg

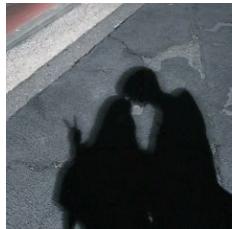


coat.jpg(<https://x.gd/ksungy>)

sea.jpg



camera.jpg(©私)



kage.jpg(<https://jp.pinterest.com/pin/763782418035482440/>)



gayoshi.jpg



text_Disney.jpg(©私)

♥ 1 "YEAR" ♥
ANNIVERSARY

► 202X.**.**

text_1st.jpg(©私)



illumi.jpg

(<https://www.pakutaso.com/20191256343post-24657.html>)



stamp1.jpg, stamp2.jpg

4. 使用した class

①既存の class

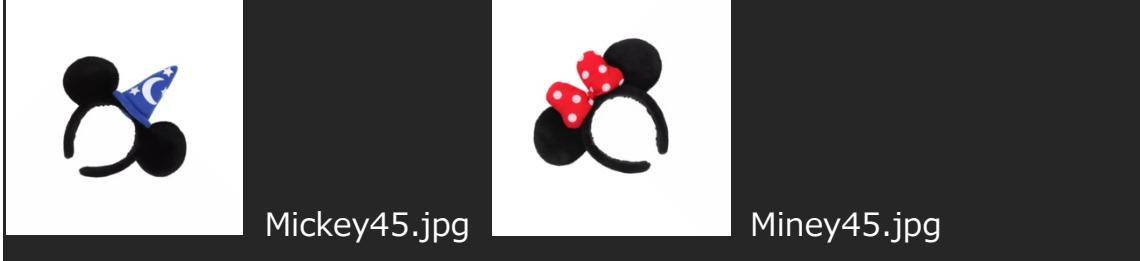
CvMain (imageProcessing1 および imageProcessing2)
Nagative
SpaceFiltering
Scale
Kmeans
Chromakey
VirtualStudio
MyImage
JpegFileWriter
JpegFileReader

②自作の class

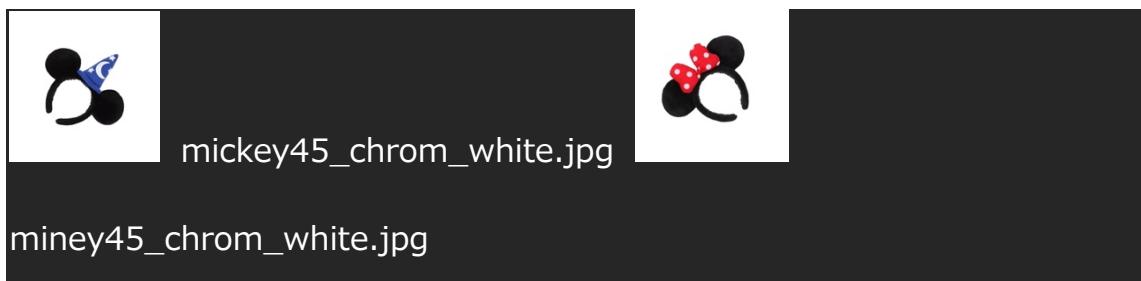
	説明
AlphaBlending	画像を重ね合わせる際アルファブレンディングするクラス
Black_background	指定の大きさの黒い画像を生成するクラス
Blending1	2枚の画像を重ね合わせたときに白でない方の画素を選び、どちらも白の場合のみ白にするクラス
Blending2	2枚の画像を重ね合わせたときに黒でない方の画素を選び、どちらも黒の場合のみ黒にするクラス
Chromakey_black_original	原画像の特定の色をもつ画素に黒を代入し、それ以外の画素は元画像のままにするクラス
Chromakey_white_original	原画像の特定の色をもつ画素に白を代入し、それ以外の画素は元画像のままにするクラス
CircleCrop	中心から円状に画像を切り取り、切り取られた部分には白を代入するクラス
Frame	画像の周りに白いフレームを生成するクラス
GammaCorrection	各画素の RGB 値を補正するクラス
ImageCropper	指定の大きさに画像を左上を起点に切るクラス
ImageCropper2	指定の大きさに画像の中心を切り取った後の画像も中心となるよう起点を計算し切るクラス
ImageCropper3	指定の大きさに画像の指定した点を起点に切るクラス
Photoframe	チエキ風の白いフレームを生成するクラス
Rotation	画像を回転させるクラス、rotate90, 180, 270, 180scale2, 45Right, 45Leftのメソッドがある。
Skyimage	上方が暗く下方が明るいグラデーションのある夜空を生成するクラス
Tiling	タイル状に画像を横に並べるクラス
Tiling2	タイル状に画像を縦に並べるクラス
GifSequenceWriter	複数枚のjpeg画像をgif画像にするクラス

5.加工の手順

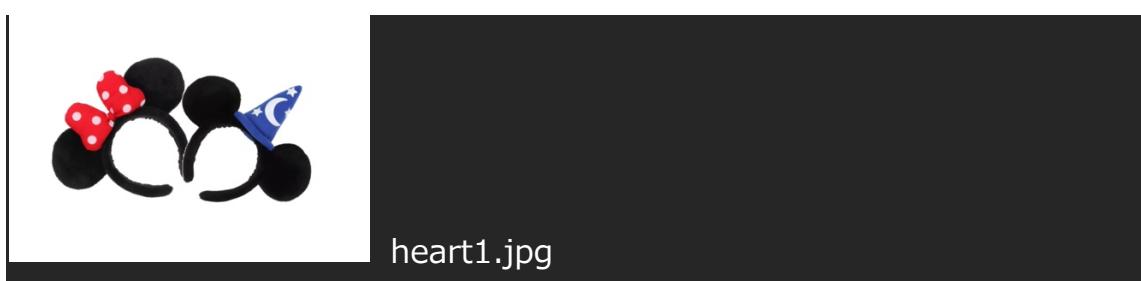
5.1. Rotation.java の rotate45Left と rotate45Right を使い、ミッキーとミニーのカチューシャを 45 度回転させる。



5.2. Chromakey_white_original.java をつかってクロマキー処理をし、背景を白とした画像を作成。



5.3. Blending1.java で任意の場所で重ね合わせたときに、画像が重なった場合白でない方の画素を選び、どちらも白の場合のみ白にするようにプログラムでカチューシャをハートの形にする。



5.4. Skyimage.java を用いて夜空を生成するプログラム 上の方が暗く下の方が明るくなるようにグラデーションをつける。



5.5. disney.jpg を Chromakey_black_original.java でクロマキー処理をし、背景黒として城だけ抽出。



disney2.jpg

5.6. disney2.jpg と night_sky.jpg を Blend2.java 任意の場所で重ね合わせたときに、画像が重なった場合黒でない方の画素を選び、どちらも黒の場合のみ黒にするように合成して、夜のディズニー画像の生成。



disney_night.jpg

5.7. right.jpg に Chromakey_black_original.java をつかって、背景黒の画像を作成。



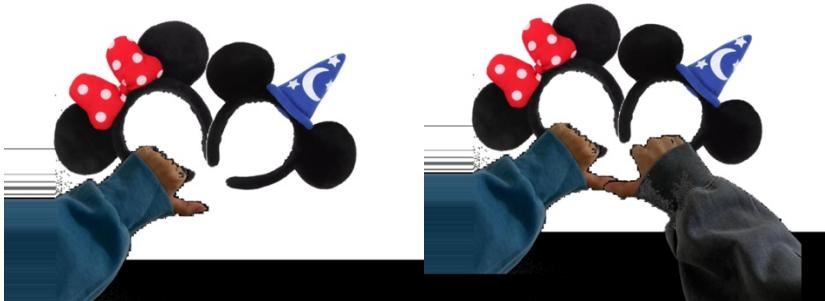
right2.jpg

5.8. left.jpg に Chromakey_black_original.java をつけて、背景黒の画像を作成。



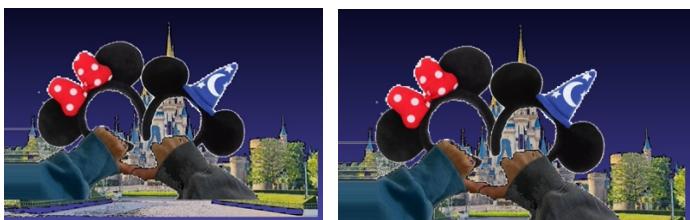
left2.jpg

5.9. right2.jpg と left2.jpg の大きさを Scale.java で調整し、Blend2.java で heart1.jpg と合成する。その後いらないところを ImageCropper.java で切り取る。



hands_and_heart.jpg

5.10. hands_and_heart.jpg と disney_night.jpg を Blend1.java で合成して night_and_heart.jpg を作成。その後いらないところを ImageCropper.java で切り取る。



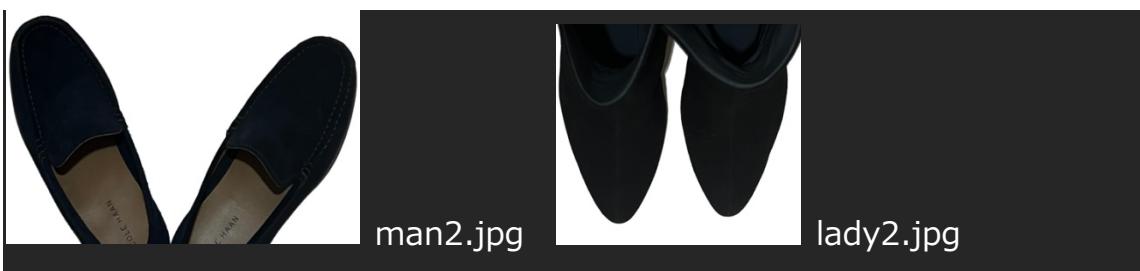
night_and_heart.jpg

5.11. Photoframe.java でチエキのようとするプログラムで画像にフレーム追加。

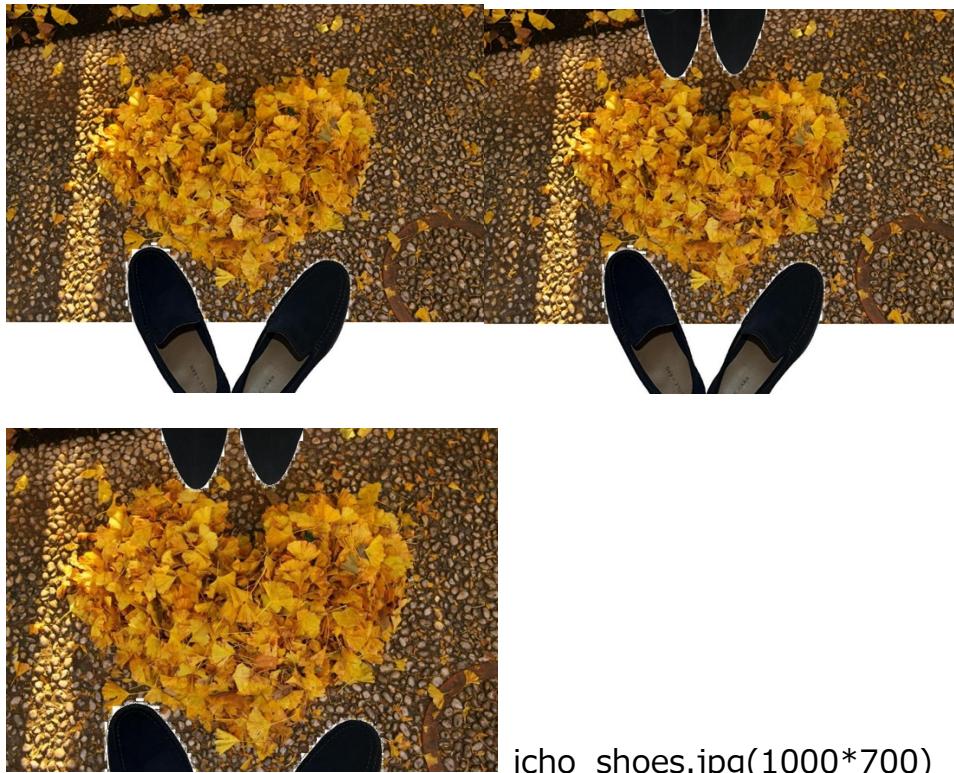


cheki.jpg(1540*1925)

5.12. man.jpg と lady.jpg を背景白のクロマキー処理をする。その際 lady.jpg は
Rotation.java で 180 度回転させる。



5.13. man2.jpg と lady2.jpg を icho.jpg に Blend1.java で合成する。
ImageCropper2.java で中心から 1000*700 の大きさに切り出した。



5.14. car.jpg を指定の位置から ImageCropper3.java を用いて 1000*700 の大きさに切り出した。

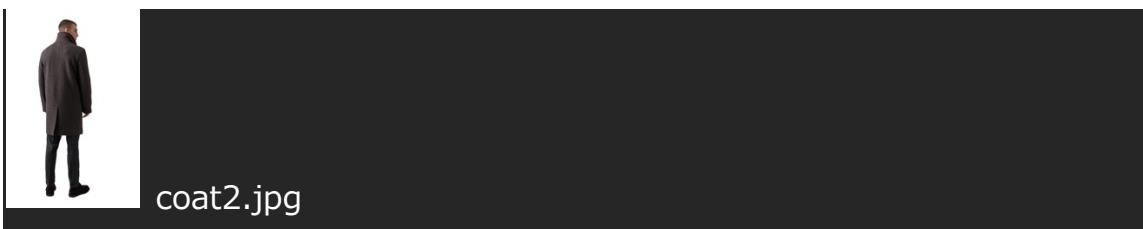


5.15. darts.jpg を Scale.java と ImageCropper.java を用いて 1000*700 の大きさに切り出した。

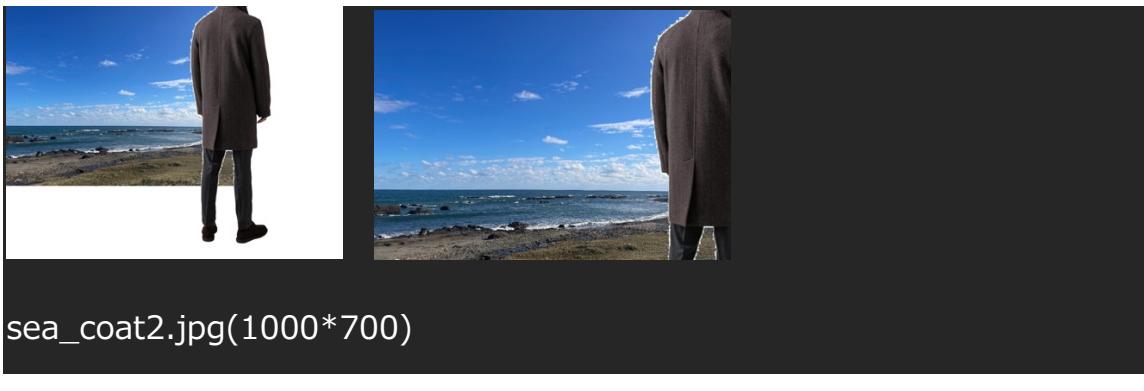


darts2.jpg(1000*700)

5.16. coat.jpg に Chromakey_white_original.java をつかって、背景白の画像を作成。



5.17. coat.jpg と sea.jpg を Blend1.jpg で合成し、ImageCropper.java で切り取る。

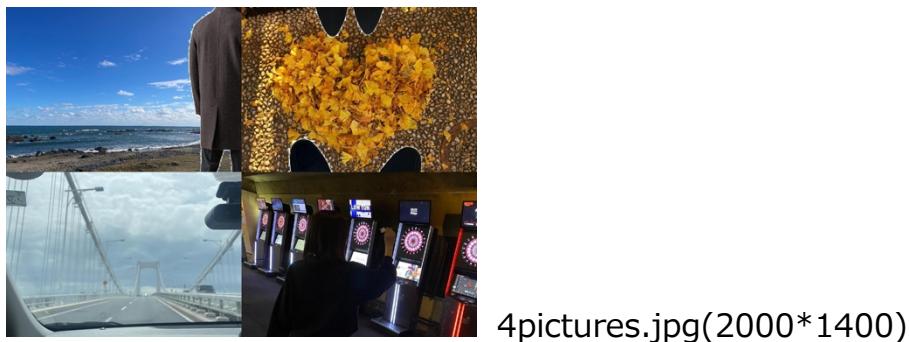


5.18. 4枚の画像 icho_shoes.jpg, sea_coat2.jpg, car2.jpg, darts2.jpg を

Tiling.java で横方向に画像を置く。



Tiling2.java で縦方向に画像を置く。



4pictures.jpg(2000*1400)

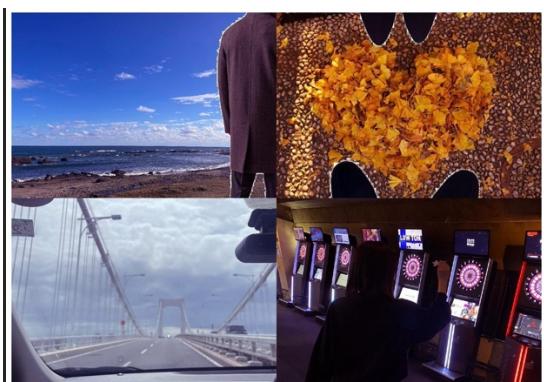
5.19. フィルムカメラ風に GammaCorrection.java の値を R = 1.0, G = 0.8, B =

1.0 に調整。



4pictures2.jpg

5.20. Frame.java で白いフレームをつける。



4pictures3.jpg

5.21. kage.jpg を円形に CircleCrop.java で切り抜く。



kage2.jpg

5.22. kamera.jpg と kage2.jpg を合成する。gayoshi.jpg と

AlphaBlending.java で合成することで画用紙感を出す。



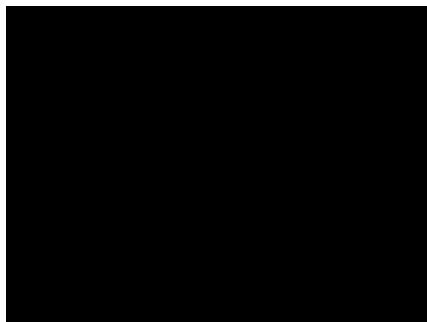
camera2.jpg

5.23. text_1st.jpg の文字の模様を Blend2.java を用いて illumi.jpg にする。



text_1st2.jpg

5.24. Black_background.java で背景の真っ黒の画像を生成する。



background.jpg

5.25. gayoshi.jpg と background.jpg を AlphaBlending.java で合成し、画用紙
感を出す。



background2.jpg

5.26. 作成した画像を Blend1.java で配置していく。



5.27. その他 stamp1.jpg や stamp2.jpg をアルバムでいうシールのように

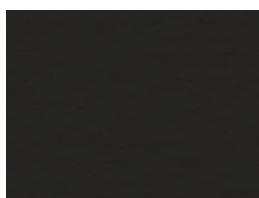
Blend1.java で装飾していく。



album.jpg

完成！！

5.28.おまけの表紙作成



background2.jpg



colorful.jpg(©私)

Memory

text_memory.jpg(©私)

文字の模様を変え、背景の黒画用紙に合成した。文字は細かい分クロマキー処理が粗い

のが改善点の余地がありそうだ。



text_memory4.jpg

<https://github.com/Deep-Symmetry/wayang/blob/main/src/main/java/org/deepsymmetry/GifSequenceWriter.java>

jpg から gif にするためのクラス。上のサイトを参考にした。



6. 頑張った点・反省点

自分の手の写真を撮影する際、背景にカーテンやファイルなど、グリーンバックに適したものを使
用意したが、iPhone の性能が優れているため、背景を均一に整えるのが k-means を使
っても難しかった。特に白壁を背景にする際、その模様を認識しない程度に影が入らないよ
う調整するのが大変だった。画像を計画通りの大きさに調整するために、Scale.java を使
用して倍率を計算し、座標を考慮しながら画像を配置した。クロマキー処理がまだ粗いた
め、近くで見ると境界線が残ってしまうことがあるので、今後はもっと精度よく切り抜けるように
工夫したい。