



情報システム基礎演習

2. 画像処理(2)

ビットプレーン統合

【コード 7】 program21.pyに、1st bit planeにペア学生の1st bit plane画像、2nd bit planeに、コード 7 で読み取った1st bit plane画像をそれぞれ埋め込み、画像を再構成するプログラムを完成させましょう。

ヒント：OR演算は $|$ 記号で表します。

【課題 3 - 2】 コード 7 で再構成した画像を再びビットプレーン分解し、各ビットプレーンを保存しなさい。また、1st bit planeや2nd bit planeが書き換わっても元の画像の見た目にあまり影響がないことを確認しなさい。

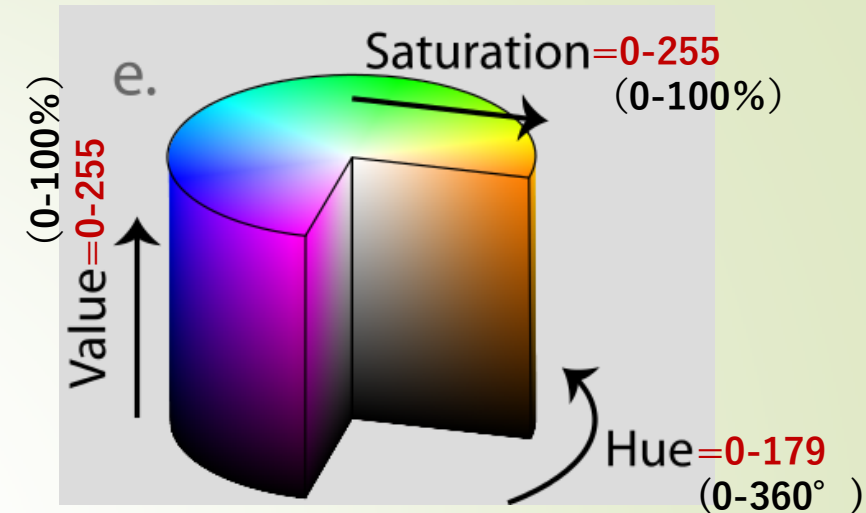
【課題 3 - 3】 8bit planeと7bit planeのみ（残りのビットはすべて0）の画像を生成し、元画像と比較し、上位ビットのみを残しても、必ずしも画像が正しく再構成できないことを確認しなさい。

HSV色空間

- 色を構成しているもの
 - Hue 色相（何色か）
 - Saturation 彩度（鮮やかさ）
 - Value 明度（明るさ）
- BGR色空間をHSV色空間に変換する＝類似した色をまとめて扱うことができる

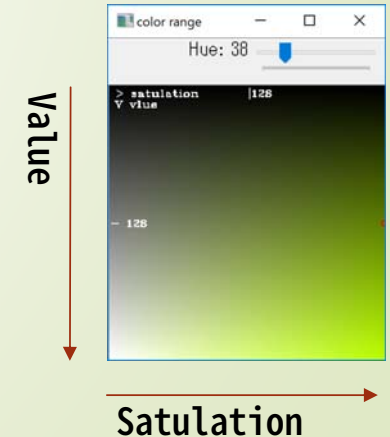
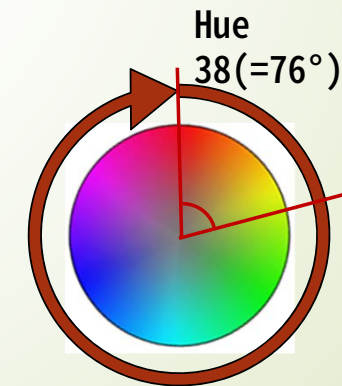
`cv2.cvtColor(img, cv2.COLOR_BGR2HSV)`

【コード 8】 program22.pyをhbar.bmp, sbar.bmp, vbar.bmpに対して実行し，各画像の四角形の中の色のBGR,HSVの値をまとめましょう。



Hueでは，何色であるかは，角度のみで決まる

Hueが一定(38: $38 \times 2 = 76^\circ$)であれば，Saturationが上がれば鮮やかに，Valueが上がれば明るくなる



クロマキー合成

- クロマキー（Chroma Key）合成とは、色度（Chroma）を手掛かりにして行う画像の合成法



グリーンバック画像



背景素材画像

マスク（＝特定の部分のみを表示（抽出）し、それ以外の部分を表示しない画像）



- グリーンバック画像内の緑を検出して黒、それ以外を白に置き換えたマスク画像



クロマキー合成画像

マスクが白の部分はグリーンバック画像を残し、黒の部分は背景素材画像に置き換えた画像

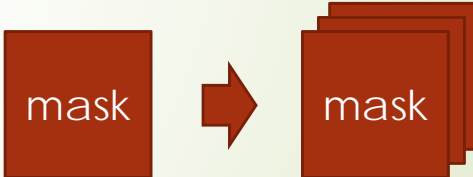
クロマキー合成（静止画）

1. program22.pyを用いて、zeni23.pngの背景色のHSV範囲を調べる
ヒント：グリーンバックは，モノや背景の影，環境光や反射光の乱れ等，様々な要因で一定にはならない → HSVに幅を持たせる（上限と下限を設定する）
2. 1の結果をもとにマスクを作成

```
mask = cv2.inRange(hsv, (h_min, s_min, v_min), (h_max, s_max, v_max))
```


mask: 挟まれた色の画素は255, それ以外は0の2次元配列
ヒント：マスクがある程度うまく作成できるまでパラメータを調整する
3. マスクを3次元配列に拡張（コメント#を外す）

```
mask = np.tile(mask[:, :, None], [1, 1, 3])
```


マスク画像を複製して，3チャンネルにする

4. クロマキー合成をする（コメント#を外して編集をする）
ヒント：①前景（zeni23.png）と背景は同サイズ，ドンネル数である
②Pythonでは，画像行列のうち，特定の条件を満たすものだけ抽出する方法がある
ex)

```
img[mask==255]
```

 # img画像のうち，maskの値が255となる「位置」のimgの画
素値

クロマキー合成（静止画）

【コード 9】 program23.pyを完成させましょう。

【課題 4】 コード 9 の結果をもとにクロマキー合成の結果についてまとめましょう。

参考：「ガチャピン クロマキー」で検索してクロマキー合成の問題点について考えてみましょう。 ※権利上レポート等に検索結果画像は使ってはいけません

【発展課題】 program23.pyの背景や前景（グリーンバック画像）を変えてみましょう。
※ただし、前景と背景は同じサイズでないとエラーになります。

動画を扱う

動画画像（≡カメラ画像）

- セルアニメーションと似ている

- 一定間隔で並んだ静止画の列

⇒ 1枚ごとの静止画

= フレーム

⇒ フレームの間隔

= フレームレート

= FPS (Frame / Sec)

fpsが大きいほどこの間隔は狭くなる



フレーム

カメラや動画がオープンの間（無限ループ）

フレームを読み込む

読み込んだフレームを表示

開いているウィンドウを破棄
カメラを解放

#[program24.py] もっとも単純な動画表示サンプル

-*- coding: utf-8 -*-

import cv2

import numpy as np

ファイル名→デバイス番号にするとカメラ起動
Ex) cv2.VideoCapture(0)

main-----

def main():

cap = cv2.VideoCapture("./mymovie.mp4")

fps = 15 #サンプル動画のFPSは15

#空のウィンドウを生成

cv2.namedWindow("video player", cv2.WINDOW_AUTOSIZE)

while cap.isOpened() :

ret, frame = cap.read()

1000/fps msecキー入力を待つ
qが押されるとwhileを抜ける

if cv2.waitKey(int(1000/fps) & 0xFF == ord('q'))

or ret == False:

break

cv2.imshow("video", frame)

cv2.destroyAllWindows()

cap.release()

run-----

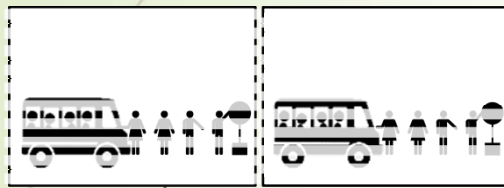
if __name__ == '__main__':

main()

動画を扱う

- ▶ (蛇足) テレビ信号で使われているのは 30FPS

ただし、2フレームで1画面（フィールド）となるインタレース方式（1フレームごとに半分ずる送信） ⇔ プログレッシブ方式



インタレース方式
2フレームで
1フィールド
(1つの画像)

- ▶ program24.pyを実行して動画ファイルを再生してみましょう。

①FPSの値を変更してみましょう。

→ FPSと動きの違い（カクつく）

→ FPSは早ければよい？（スピード感）

②PCのカメラを使ってみましょう。

→カメラのときにもFPSを変更してみましょう（値を大きくしてみましょう）

クロマキー合成（動画）

【コード 10】 program23.pyのコードを参考に，program25.pyを完成させましょう．

【発展課題】 program25.pyの背景や前景（グリーンバック画像）を変えてみましょう．
※ただし，前景と背景は同じサイズでないとエラーになります．

参考：クロマキー合成はフィルム時代から現在の実写とCGの融合映像まで，映画やテレビ作品などでよく用いられてきました．バックの色には青や緑がよく用いられます（理由は諸説あります）．

今回は合成の境界部分やバック検出漏れノイズ，欠けに対する対策はしていないため，きれいな合成にはなっていません．興味があれば調べてみましょう．

レポート課題

【小レポート】実験記録のWordファイルを完成させる。

【大レポート】実験記録と結果のファイルをもとに以下のレポートをまとめる。

1. はじめに …デジタル画像とその色表現について触れ、目的を述べる
2. 実験環境 …pythonとOpenCVについてまとめる
3. 実験内容 …色ヒストグラム、フィルタ、ビットプレーン、クロマキーについて、原理と内容の説明（結果には触れない） ※命令文やプログラムコードの説明は不要
4. 結果 …実験結果についてまとめる（考察はしない）
5. 考察 …【課題1】～【課題4】を参考に、結果について考察する
6. おわりに …各実験の結果からデジタル画像処理についての考えをまとめる（感想や実験の成否は「書かない」）

参考文献 …[番号]著者，タイトル，書名，ページ，出版社（発行年）
※Webは原則参考文献としないが（**Wikipedia厳禁**），どうしても場合は，
[番号]（著者，）タイトル，URL（作成年）

動画像処理とクロマキー（動画）は除く

レポート注意事項

- A4用紙左詰めで作成し、印刷時は表紙を付け、左2か所ホッチキス留めする
Wordを用いた場合、オートインデント等により本文が左詰にならないことがあることに注意
- 内容は必ず「文章」で書き、内容ごとに章・節・項に分類し、「〇.〇.〇 タイトル」を付ける（章節項のタイトルに該当しないタイトルや文章以外の単語は書いてはいけない）
- 箇条書きだけの章、節、項を作ってはならず、本文でどのような箇条書きか説明しなくてはならない
- 「自分で行った実験」を報告するレポートのため、「〇〇せよ」等の指示表現を用いてはならない
- 図表には番号、タイトルをつけ本文で説明しなくてはならない（番号は、レポートで通し、または「章」ごと.） ex) 図1,2,⋯ または 1章の図なら 図1.1, 1.2,⋯
- 参考文献の章には章番号を付けない、テキスト、Web資料は参考文献にしてはならない
- 参考文献は番号を付し、本文の参考箇所にも番号を示す（本文で参考のない文献は載せない）
なお、同じ文献を複数個所で参考にする場合は番号を1つにまとめる
ex) 〇〇といわれている[1]. ⋯△△である[1].