

# 笑顔の写真を自動で選び出すサイト制作

情報メディア学科 斎藤一ゼミナール

1821200 原本雅人

# 目的・ターゲット

## ターゲット

写真を整理したい

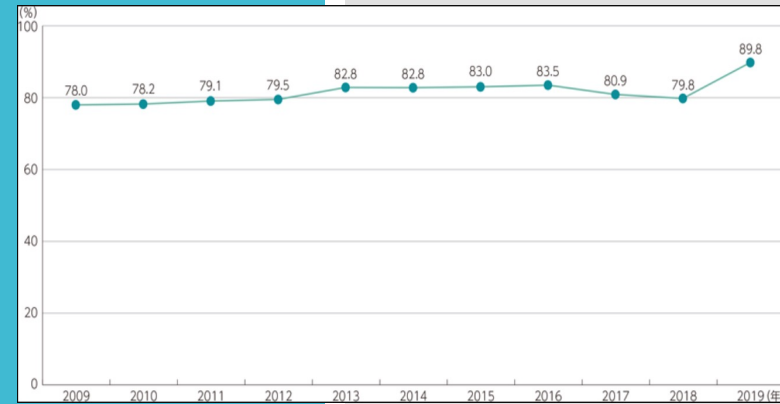
1. 小さなお子さんのいる世代
2. イベントの幹事

## 目的

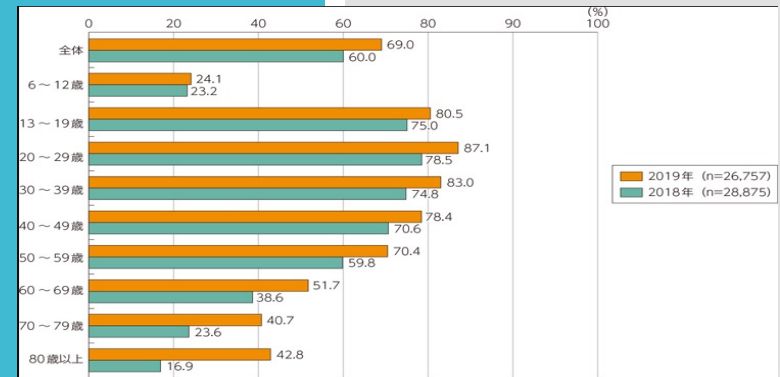
笑顔の写真だけを選び出したい

# 背景1

近年、インターネットの利用率が高い[1]



SNSの利用率が高い[2]



[1][2]総務省 | 令和2年版 情報通信白書 | インターネット利用状,  
<https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/ro2/html/nd252120.html>

# 背景1

Instagram の利用者数が急伸[3]

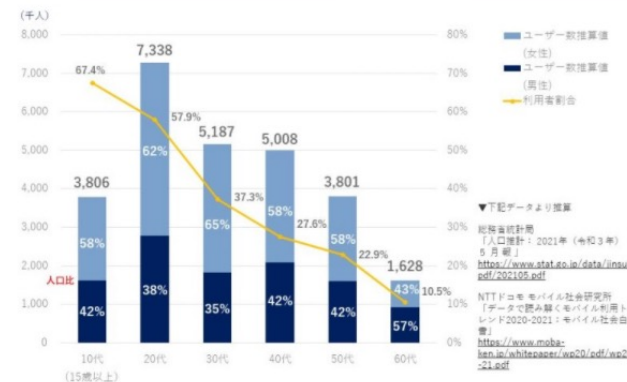
若者の利用者数が多い[4]

- 写真映えする画像を求める
- 写真を気軽に投稿できる

国内主要SNS | MAU数の推移 (年別)



Instagramの年齢別ユーザー数 男女別 (国内)



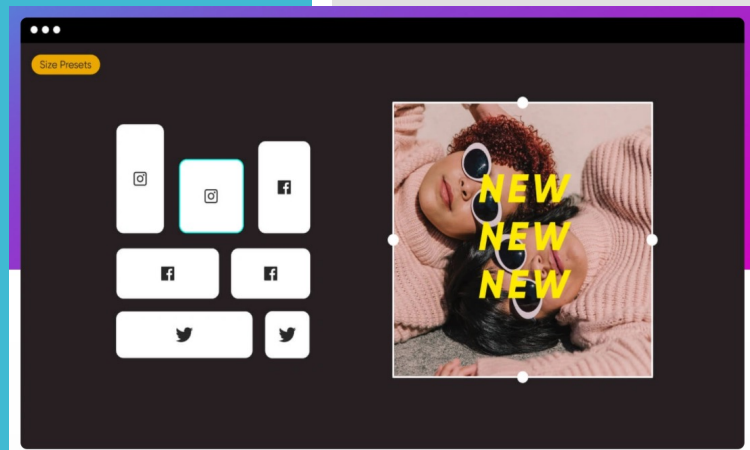
[3],[4]2017年7月更新。12のソーシャルメディア最新動向データまとめ、<https://gaiax-socialmedialab.jp/post-30833/>

# 先行事例1

写真の加工アプリ  
PicsArt, Adobe Photoshop,  
etc...



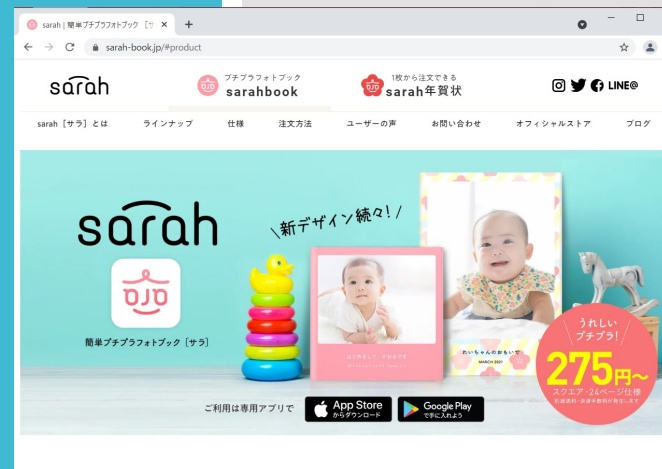
アプリのインストール  
に手間がかかる



# 先行事例2

フォトアルバムを作るサイト・アプリ  
sarah (サラ), ビスタプリント, etc...

写真をまとめられる

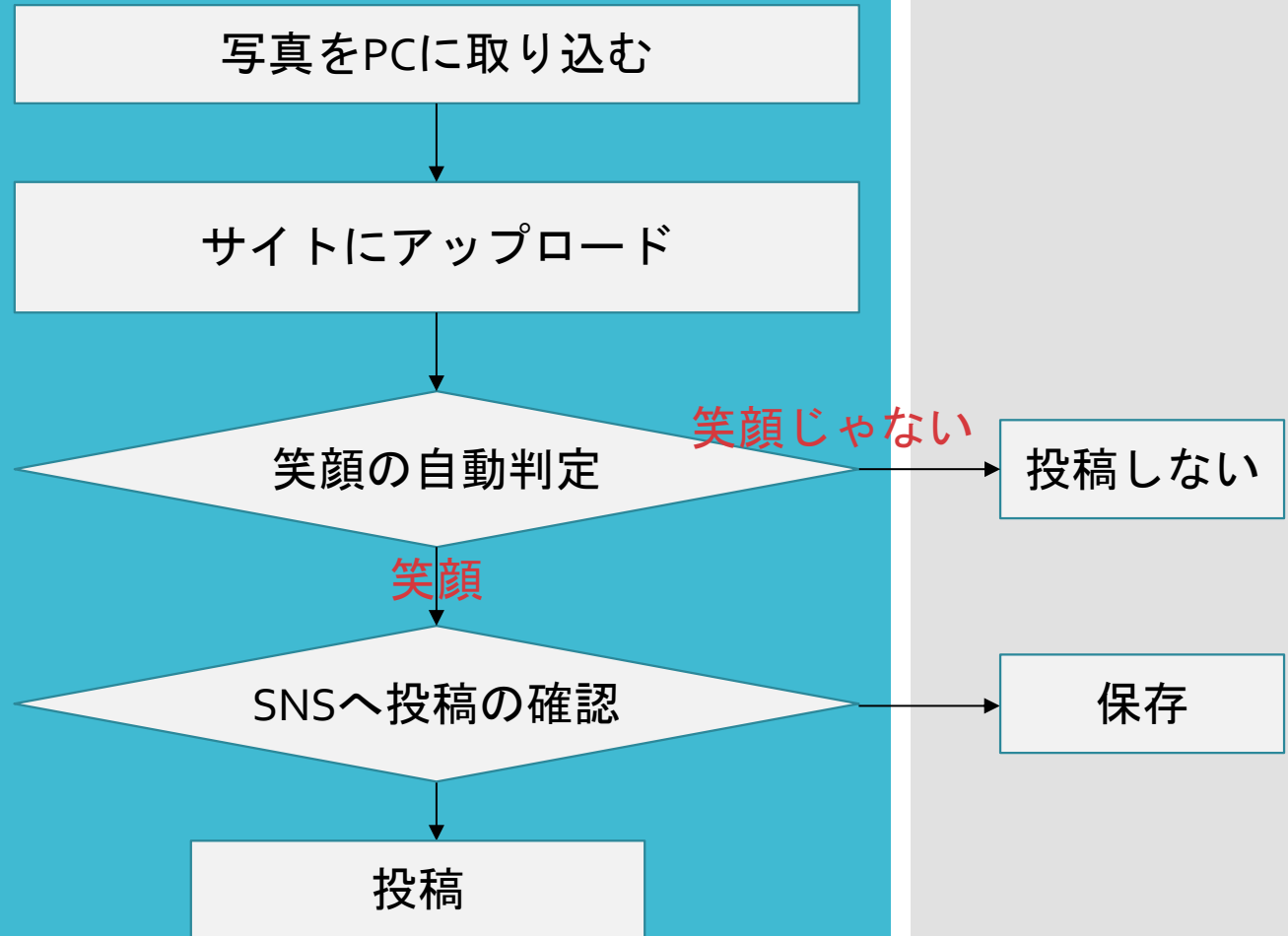


# 考察・制作テーマ

アプリをインストールせず、  
Webサイトで気軽に写真の選別がしたい

笑顔の写真を自動で選び出してくれるサイト

# 運用・流れ





# 使用技術

- HTML (Web)
- CSS (Web)
- Python (OpenCVのベース)
- OpenCV
  - 画像取得
  - 顔認識
  - 笑顔認識

```
SmileDetect.py

import cv2

capture = cv2.VideoCapture(0)
capture.set(3,640)# 320 320 640 720
capture.set(4,360)#180 240 360 405

face_cascade = cv2.CascadeClassifier('./haarcascade_frontalface_default.xml')
smile_cascade = cv2.CascadeClassifier('./haarcascade_smile.xml')

while True:
    ret, img = capture.read()
    img = cv2.flip(img,1)#鏡表示にするため。
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

    faces = face_cascade.detectMultiScale(gray, 1.1, 5)
    for (x,y,w,h) in faces:
        cv2.circle(img,(int(x+w/2),int(y+h/2)),int(w/2),(255, 0, 0),2) # blue

    roi_gray = gray[y:y+h, x:x+w] #Gray画像から、顔領域を切り出す。
    smiles= smile_cascade.detectMultiScale(roi_gray,scaleFactor= 1.2, minNeighbors=
    if len(smiles) >0 :
        for (sx,sy,sw,sh) in smiles:
            cv2.circle(img,(int(x+sx+sw/2),int(y+sy+sh/2)),int(sw/2),(0, 0,
```

# スケジュール

7月	8月	9月	10月	11月	12月	1月
	調査・勉強					
			制作			
				評価・改善		
					論文執筆	

# まとめ・今後の課題

## まとめ

子供のいる家庭、イベントの幹事を対象とした笑顔の写真を選び出すサイト制作

## 課題

- ・制作において必要な知識を学ぶ
- ・内容の細かい部分の修正
- ・内容の不足部分の調査とサイト制作