

HANOI UNIVERSITY OF CIVIL ENGINEERING

Đồ án Xử lý ảnh

Đề tài: Nhận diện khôn mặt

Nguyễn Thu Huyền
Hoàng Đình Vinh

Nguyễn Minh Hoàng
Trần Anh Dũng



VÌ SAO CẦN NHẬN DIỆN KHUÔN MẶT?



- Khuôn mặt đóng vai trò quan trọng trong quá trình giao tiếp giữa người với người, nó mang một lượng thông tin giàu có, chẳng hạn như từ khuôn mặt chúng ta có thể xác định giới tính, tuổi tác, chủng tộc, trạng thái cảm xúc, đặc biệt là xác định mối quan hệ với đối tượng

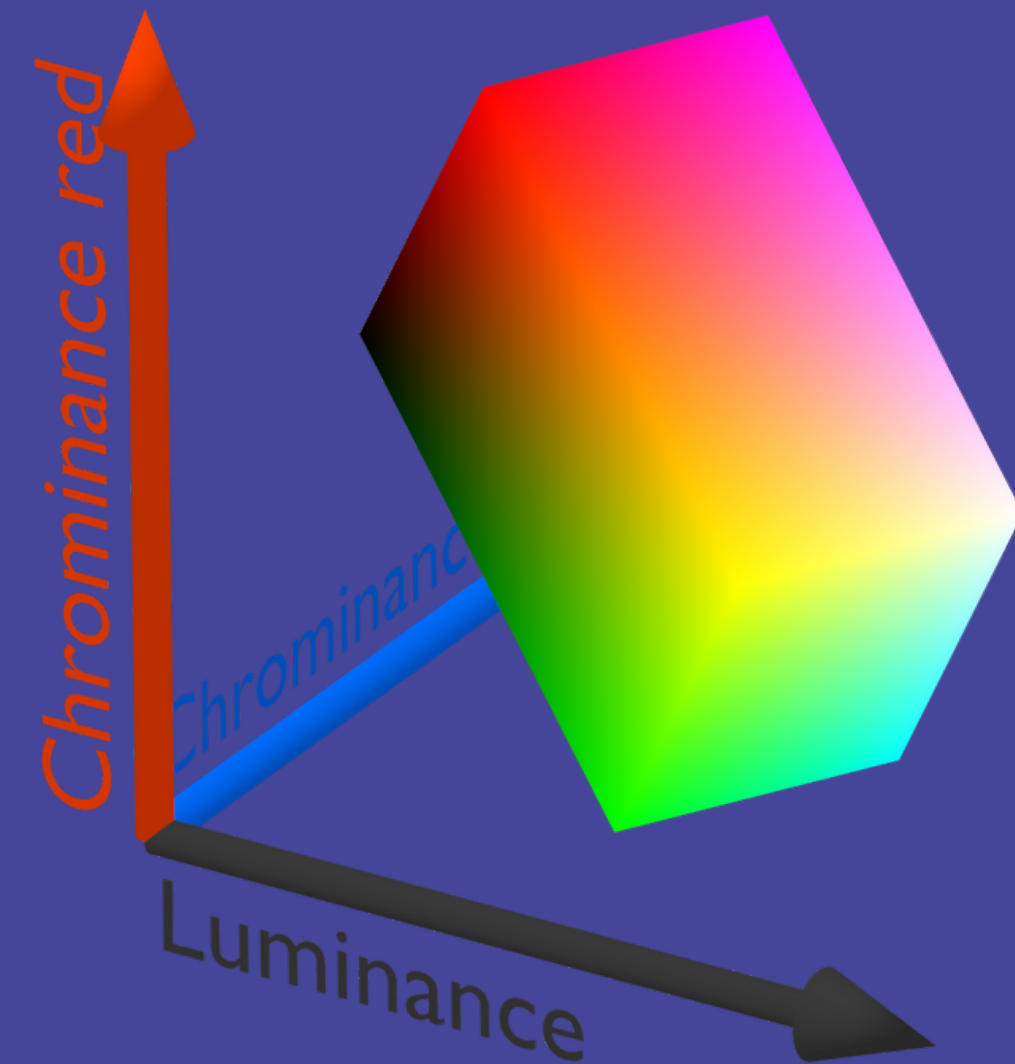
→ Do đó, bài toán nhận dạng khuôn mặt đóng vai trò quan trọng trong nhiều lĩnh vực đời sống hằng ngày của con người như các hệ thống giám sát, quản lý vào ra, tìm kiếm thông tin, ...

I. CƠ SỞ LÝ THUYẾT

1. Phát hiện vùng da trên không gian màu Ycbcr

YCBCR

- Phát hiện vùng da trên không gian màu YcbCr là một kỹ thuật xử lý ảnh được sử dụng để xác định các vùng trong một hình ảnh có khả năng chứa da người.
- Không gian màu Ycbcr là một không gian màu phổ biến trong xử lý ảnh, được sử dụng trong nhiều ứng dụng như truyền hình, video



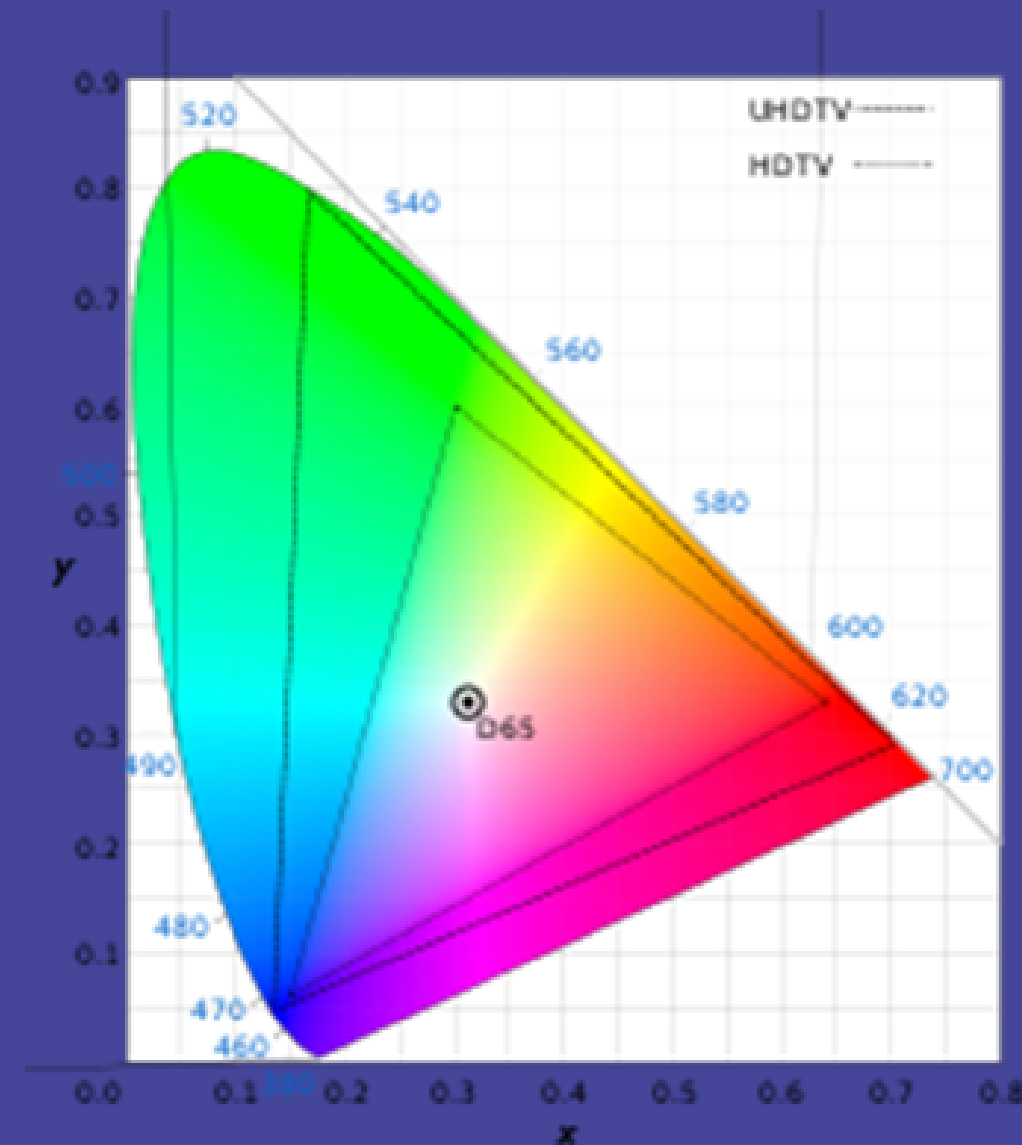
YCBCR

- Trong không gian màu này, các giá trị RGB của mỗi pixel được chuyển đổi thành các giá trị Y, Cb, Cr như sau:

$$Y = 0.299R + 0.587G + 0.114B$$

$$Cb = -0.1687R - 0.3313G + 0.5000B$$

$$Cr = 0.5000R - 0.4187G - 0.0813B$$



YCbCr

Các đặc tính khác biệt của da người trong không gian màu YCbCr bao gồm :

- Giá trị Y cao: Da người có giá trị Y cao hơn đáng kể so với các vật thể khác trong môi trường
- Giá trị Cb thấp: Da người có giá trị Cb thấp hơn đáng kể so với các vật thể có màu xanh lá cây hoặc xanh dương.
- Giá trị Cr thấp: Da người có giá trị Cr thấp hơn đáng kể so với các vật thể có màu đỏ hoặc tím.

YCBCR

Xác định giới hạn da người

ta xác định được lower bound (giới hạn dưới) là $[0, 133, 77]$ và upper bound (giới hạn trên) là $[255, 173, 127]$ của màu da trong không gian màu YCbCr.

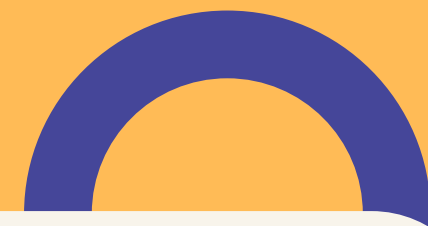
Original Image



YCbCr Image



I. CƠ SỞ LÝ THUYẾT

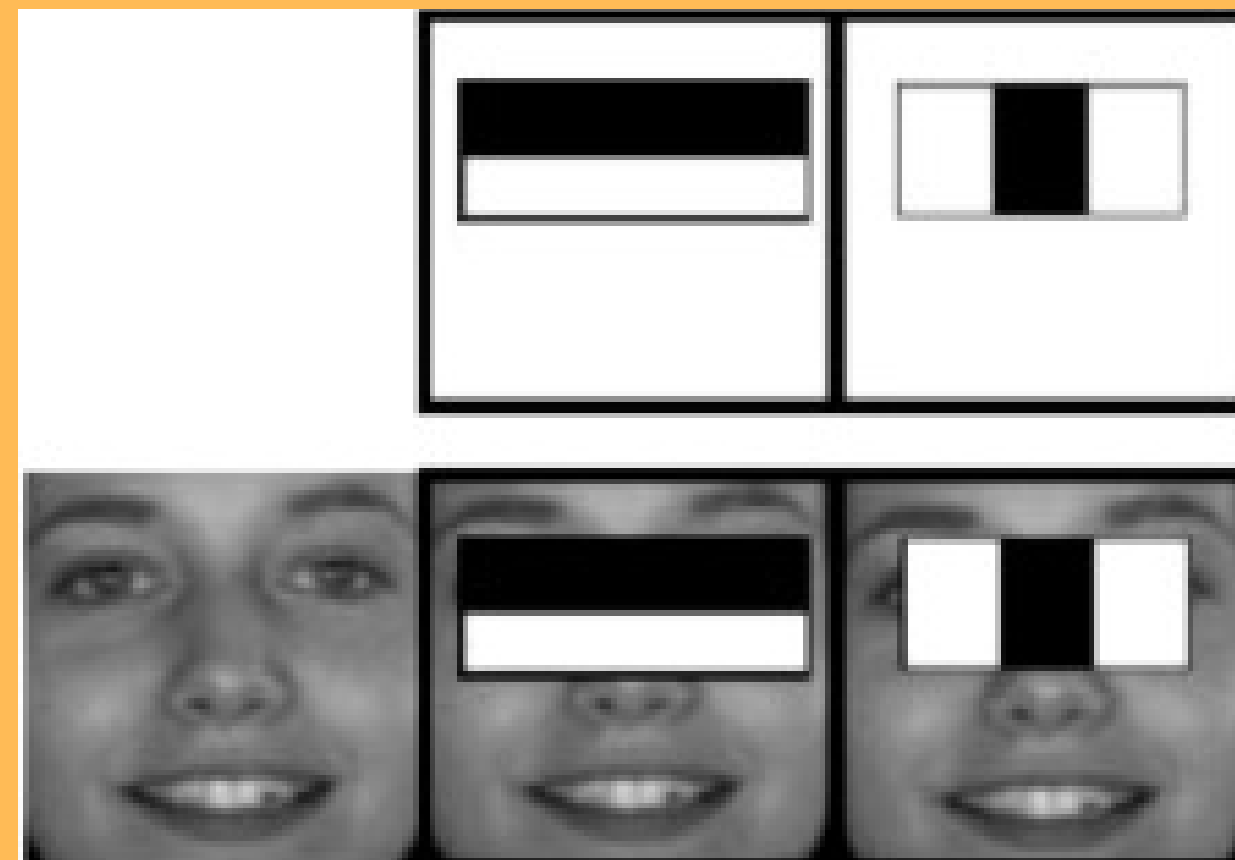


2. Haar cascade

KHÁI NIỆM CƠ BẢN

Haar cascade

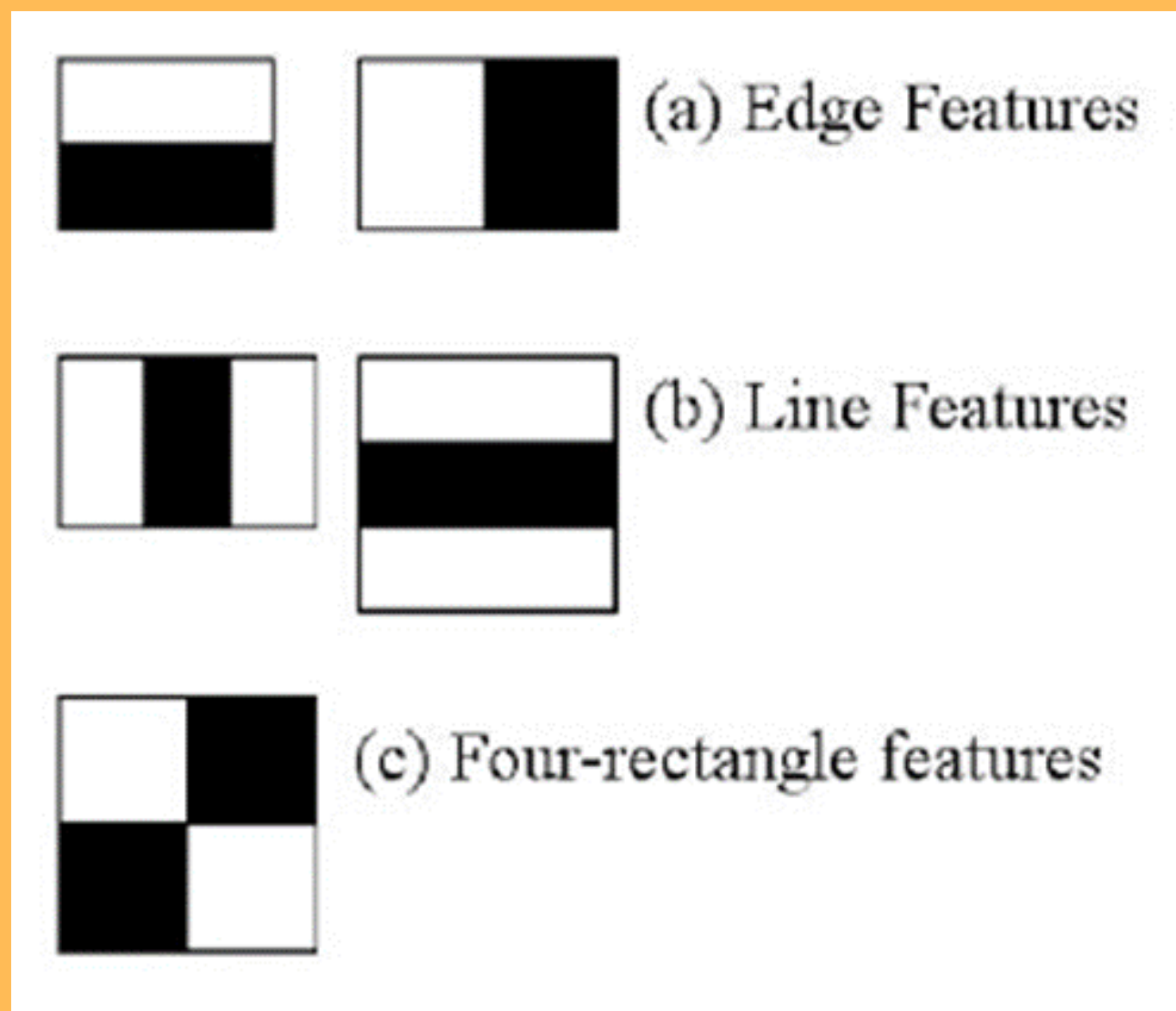
Haar cascade là một thuật toán phát hiện đối tượng được giới thiệu bởi Paul Viola và Michael Jones vào năm 2001. Nó hoạt động bằng cách sử dụng các đặc trưng Harr để xác định các đối tượng trong hình ảnh.



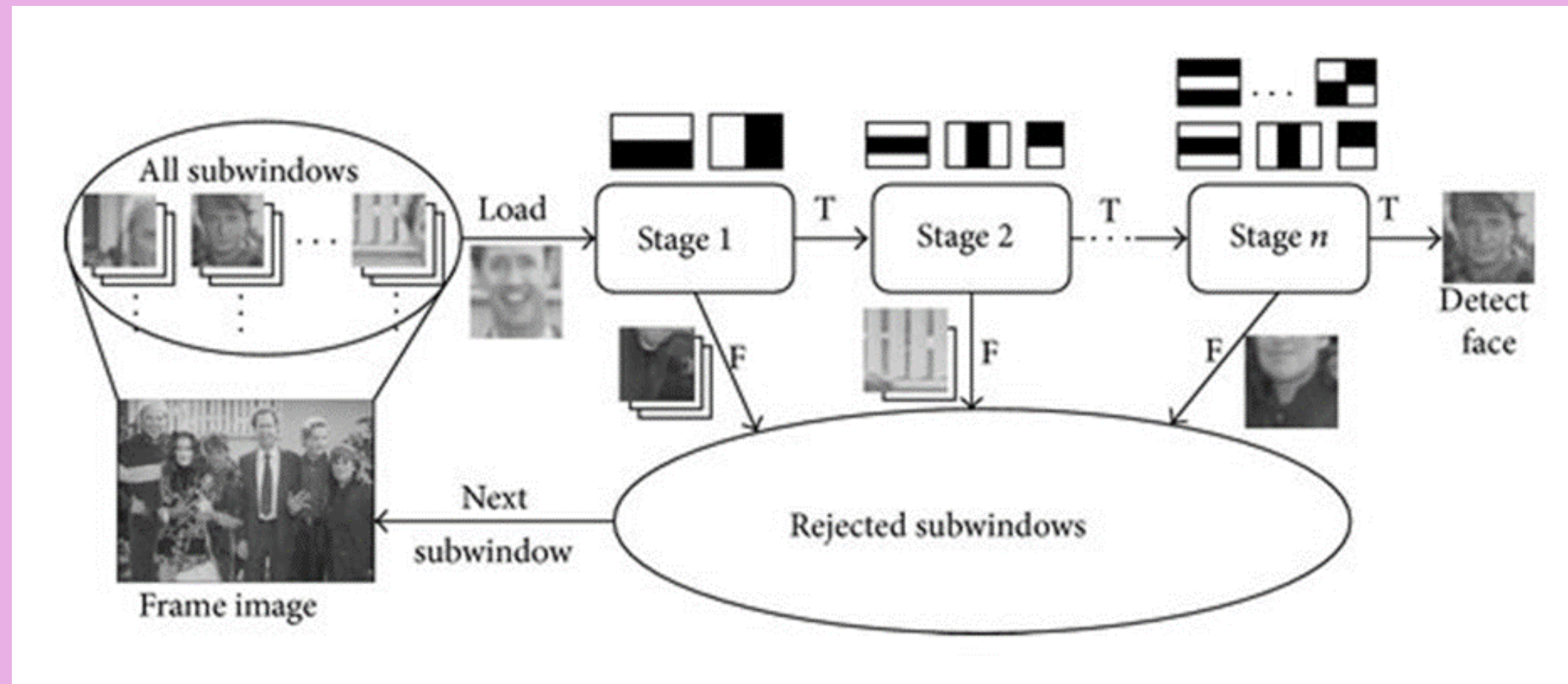
KHÁI NIỆM CƠ BẢN

Haar cascade

Các đặc trưng Haar là các tính toán toán học đơn giản được thực hiện trên các vùng nhỏ của hình ảnh. Chúng có thể được sử dụng để phát hiện các đặc điểm đặc trưng của khuôn mặt, chẳng hạn như mắt, mũi, miệng, vùng sáng và tối ...



CÁC BƯỚC NHẬN DIỆN BẰNG HAAR CASCADE



- **Bước 1:** Hình ảnh (đã được gửi đến bộ phân loại) được chia thành các phần nhỏ (hoặc các cửa sổ con như trong hình minh họa).
- **Bước 2:** Chúng ta đặt N không có bộ dò theo cách xếp tầng trong đó mỗi bộ phát hiện sự kết hợp của các loại đặc trưng khác nhau từ các hình ảnh được truyền qua.
- **Bước 3:** Hình ảnh (hoặc hình ảnh phụ) có độ tin cậy cao nhất được phát hiện dưới dạng khuôn mặt và được gửi đến bộ tích lũy trong khi phần còn lại bị từ chối. Do đó, Cascade tìm nạp khung hình / hình ảnh tiếp theo nếu còn lại và bắt đầu lại quá trình.

II. Ý TƯỞNG BÀI TOÁN

Problem ideas

Đào tạo 1
haar
cascade
face
detection



Đọc ảnh và
chuyển qua
không gian
YCbCr



Thực hiện
phép
bitwise-and
giữa ảnh
đầu vào và
skin_mask



Phát hiện
khuôn mặt

ĐÀO TẠO

- Positive Images: chứa những hình ảnh chứa khuôn mặt.
- Negative Images: chứa những hình ảnh không chứa khuôn mặt.

Lưu ý: Số lượng Negative Images nhiều hơn đáng kể so với Positive Images

- Annotation files: Đối với mỗi positive image tương ứng một tệp chú thích (annotation file) để chỉ định vị trí của khuôn mặt trong hình ảnh.
- Các tham số của cascade: thông tin về các tham số đã được sử dụng trong quá trình huấn luyện
- Các đặc trưng dạng Haar: Mô hình đã được huấn luyện dựa trên các đặc trưng dạng Haar, đó là các bộ lọc hình chữ nhật được sử dụng để phát hiện mẫu trong hình ảnh.
- Các giai đoạn phân loại: Bộ phân loại cascade đã được huấn luyện bao gồm nhiều giai đoạn, mỗi giai đoạn bao gồm một tập hợp các bộ phân loại yếu.
- Ngưỡng và trọng số: chứa các giá trị ngưỡng và trọng số cho mỗi bộ phân loại yếu.
- Vị trí và kích thước của đặc trưng: lưu trữ các vị trí và kích thước của các đặc trưng dạng Haar đã được chọn.


```
import cv2
import os

positive_images_path = r'C:\Users\hoang\Desktop\PROJECTS\DIP\positive_images'
negative_images_path = r'C:\Users\hoang\Desktop\PROJECTS\DIP\negative_images'

positive_annotations_file = 'positive_annotations.txt'
negative_annotations_file = 'negative_annotations.txt'

positive_samples = 'positive_samples.vec'
num_pos_samples = len(os.listdir(positive_images_path))
num_neg_samples = len(os.listdir(negative_images_path))
num_stages = 20

command = f'opencv_createsamples -info {positive_annotations_file} -vec {positive_samples} ' \
          f'-num {num_pos_samples}'
os.system(command)

command = f'opencv_traincascade -data output -vec {positive_samples} -bg {negative_annotations_file} ' \
          f'-numPos {num_pos_samples} -numNeg {num_neg_samples} -numStages {num_stages}'
os.system(command)
```

Ta có thể sử dụng file
“face_classifier_DA.xml” vừa huấn
luyện để thực hiện tác vụ phát
hiện khuôn mặt dự trên dữ liệu
mới.

```
cascade_name = 'cascade.xml'
classifier_file = 'face_classifier_DA.xml'
os.rename(cascade_name, classifier_file)
```

ĐỌC ẢNH ĐẦU VÀO VÀ YCBCR

Từ khoảng lower bound và upper bound, ta xác định một mask skin_mask và một ảnh skin, cụ thể như sau:

- Những pixel nằm trong khoảng được quy định bằng lower bound và upper bound trong không gian Ycbcr được đặt là màu trắng (255)
- Những pixel không nằm trong khoảng trên được đặt là đen (0).

→ Ta được output là một mask với những pixel có thể là da, có màu trắng và những pixel không phải da người màu đen.

ĐỌC ẢNH ĐẦU VÀO VÀ YCBCR



THỰC HIỆN BITWISE AND

Từ mask trên, có thể thực hiện phép bitwise_and giữa ảnh gốc đầu vào và skin_mask này để giữ lại toàn bộ pixel có màu trắng, loại được đa số những vùng không phải da.

→ Ta có ảnh sau:



III. THỰC THI

```

import cv2
import numpy as np
trained_path =
r'C:\Users\hoang\Desktop\PROJECTS\DIP\face_classifier_DA.xml'
face_cascade = cv2.CascadeClassifier(trained_path)

def face_detector(image_path):
    image = cv2.imread(image_path)
    ycbcr_image = cv2.cvtColor(image, cv2.COLOR_BGR2YCrCb)

    lower_bound = np.array([0, 133, 77], dtype=np.uint8)
    upper_bound = np.array([255, 173, 127], dtype=np.uint8)

    skin_mask = cv2.inRange(ycbcr_image, lower_bound,
upper_bound)
    skin_detected = cv2.bitwise_and(image, image, mask=skin_mask)

    gray = cv2.cvtColor(skin_detected, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray,
scaleFactor=1.1, minNeighbors=5, minSize=(30, 30))

    for (x, y, w, h) in faces:
        cv2.rectangle(image, (x, y), (x + w, y + h), (0, 255, 0), 2)

    output_path = 'output.jpg'
    cv2.imwrite(output_path, image)
    print(f"saved to: {output_path}")

image_path = 'TestImage.jpg'
face_detector(image_path)

```

- Khởi tạo Phân loại Cascade khuôn mặt
- Đọc ảnh từ đường dẫn tới ảnh
- Chuyển đổi ảnh sang không gian màu Ycbcr

- Tạo ra mask từ khoảng mặt định sẵn, tạo ảnh da mặt

- Phát hiện khuôn mặt

- Vẽ bounding box cho những gương mặt được phát hiện

KẾT QUẢ



Lặp qua các khuôn mặt đã được phát hiện và vẽ các hình chữ nhật xung quanh chúng trên ảnh gốc (image).

XIN CẢM ƠN ĐÃ LẮNG NGHE!