# Facial Recognition

Facial recognition is a way of identifying or confirming an individual's identity using their face. Facial recognition systems can be used to identify people in photos, videos, or in real-time.

## How does facial recognition work?

Facial recognition does not rely on a massive database of photos to determine an individual's identity — it simply identifies and recognizes one person as the sole owner of the device, while limiting access to others.

Beyond unlocking phones, facial recognition works by matching the faces of people walking past special cameras, to images of people on a watch list. The watch lists can contain pictures of anyone, including people who are not suspected of any wrongdoing, and the images can come from anywhere — even from our social media accounts. Facial technology systems can vary, but in general, they tend to operate as follows:

### Step 1: Face detection

The camera detects and locates the image of a face, either alone or in a crowd. The image may show the person looking straight ahead or in profile.

### Step 2: Face analysis

Next, an image of the face is captured and analyzed. Most facial recognition technology relies on 2D rather than 3D images because it can more conveniently match a 2D image with public photos or those in a database. The software reads the geometry of your face. Key factors include the distance between your eyes, the depth of your eye sockets, the distance from forehead to chin, the shape of your cheekbones, and the contour of the lips, ears, and chin. The aim is to identify the facial landmarks that are key to distinguishing your face.

### Step 3: Converting the image to data

The face capture process transforms analog information (a face) into a set of digital information (data) based on the person's facial features. Your face's analysis is essentially turned into a mathematical formula. The numerical code is called a faceprint. In the same way that thumbprints are unique, each person has their own faceprint.

Step 4: Finding a match

Your faceprint is then compared against a database of other known faces. For example, the FBI has access to up to 650 million photos, drawn from various state databases. On Facebook, any photo tagged with a person's name becomes a part of Facebook's database, which may also be used for facial recognition. If your faceprint matches an image in a facial recognition database, then a determination is made.

Of all the biometric measurements, facial recognition is considered the most natural. Intuitively, this makes sense, since we typically recognize ourselves and others by looking at faces, rather than thumbprints and irises. It is estimated that over half of the world's population is touched by facial recognition technology regularly.

Facial Recognition is used for a variety of purposes. These include:

- Unlocking phones
- Law enforcement
- Airports and border control
- Finding missing persons
- Reducing retail crime
- Improving retail experiences
- Banking
- Marketing and advertising
- Tracking student or worker attendance
- Recognizing drivers

## Advantages of face recognition

- Increased security
- Reduced crime
- Greater convenience

- Faster processing
- Integration with other technologies

## Disadvantages of face recognition

- Surveillance
- Scope for error
- Breach of privacy
- Massive data storage

### Load Training Data

```
import pickle

# ...

def encode_known_faces(
    model: str = "hog", encodings_location: Path = DEFAULT_ENCODINGS_PATH
) -> None:
    names = []
    encodings = []

    for filepath in Path("training").glob("*/*"):
        name = filepath.parent.name
        image = face_recognition.load_image_file(filepath)

        face_locations = face_recognition.face_locations(image, model=model)
        face_encodings = face_recognition.face_encodings(image, face_locations)

        for encoding in face_encodings:
            names.append(name)
            encodings.append(encoding)

    name_encodings = {"names": names, "encodings": encodings}
    with encodings_location.open(mode="wb") as f:
        pickle.dump(name_encodings, f)
```

```
encode_known_faces()
```

### Recognize Unlabeled Faces

```
from collections import Counter

# ...

def _recognize_face(unknown_encoding, loaded_encodings):
    boolean_matches = face_recognition.compare_faces(
        loaded_encodings["encodings"], unknown_encoding
    )
    votes = Counter(
        name
        for match, name in zip(boolean_matches, loaded_encodings["names"])
        if match
    )
    if votes:
        return votes.most_common(1)[0][0]

recognize_faces("unknown.jpg")
```

### Display Results

```
from PIL import Image, ImageDraw

# ...

def recognize_faces(
    image_location: str,
    model: str = "hog",
    encodings_location: Path = DEFAULT_ENCODINGS_PATH,
) -> None:

    # ...

    pillow_image = Image.fromarray(input_image)
    draw = ImageDraw.Draw(pillow_image)
```

```python

    for bounding_box, unknown_encoding in zip(
        input_face_locations, input_face_encodings
    ):
        name = _recognize_face(unknown_encoding, loaded_encodings)
        if not name:
            name = "Unknown"
        # Removed print(name, bounding_box)
        _display_face(draw, bounding_box, name)

    del draw
    pillow_image.show()
```

```python
BOUNDING_BOX_COLOR = "blue"
TEXT_COLOR = "white"

# ...

def _display_face(draw, bounding_box, name):
    top, right, bottom, left = bounding_box
    draw.rectangle(((left, top), (right, bottom)), outline=BOUNDING_BOX_COLOR)
    text_left, text_top, text_right, text_bottom = draw.textbbox(
        (left, bottom), name
    )
    draw.rectangle(
        ((text_left, text_top), (text_right, text_bottom)),
        fill="blue",
        outline="blue",
    )
    draw.text(
        (text_left, text_top),
        name,
        fill="white",
    )
```