Big Data and Automated Content Analysis (12EC)
Week 10: »Multimedia Data«
Wednesday

Felicia Loecherbach
f.loecherbach@uva.nl

Apr 16, 2024

UvA RM Communication Science

Today

Multimedia data

- The shift towards multimedia data
- Representing multimedia data
- Classic SML
- Deep Learning
- (Commercial) APIs

Next steps

Today: Beyond words: Multimedia Data

Multimedia data

Multimedia data

The shift towards multimedia data

Why do Images Matter?

People are more likely to pay attention to visuals

IMMIGRATION

How America Got to 'Zero Tolerance' on Immigration

Battles have raged within the White House over family separations, ICE raids and President Trump's obsession with a wall.

Together, they have remade homeland security.

15m ago 393 comments

Mr. Trump's approach follows a model from Europe and Australia, our Interpreter columnists write.

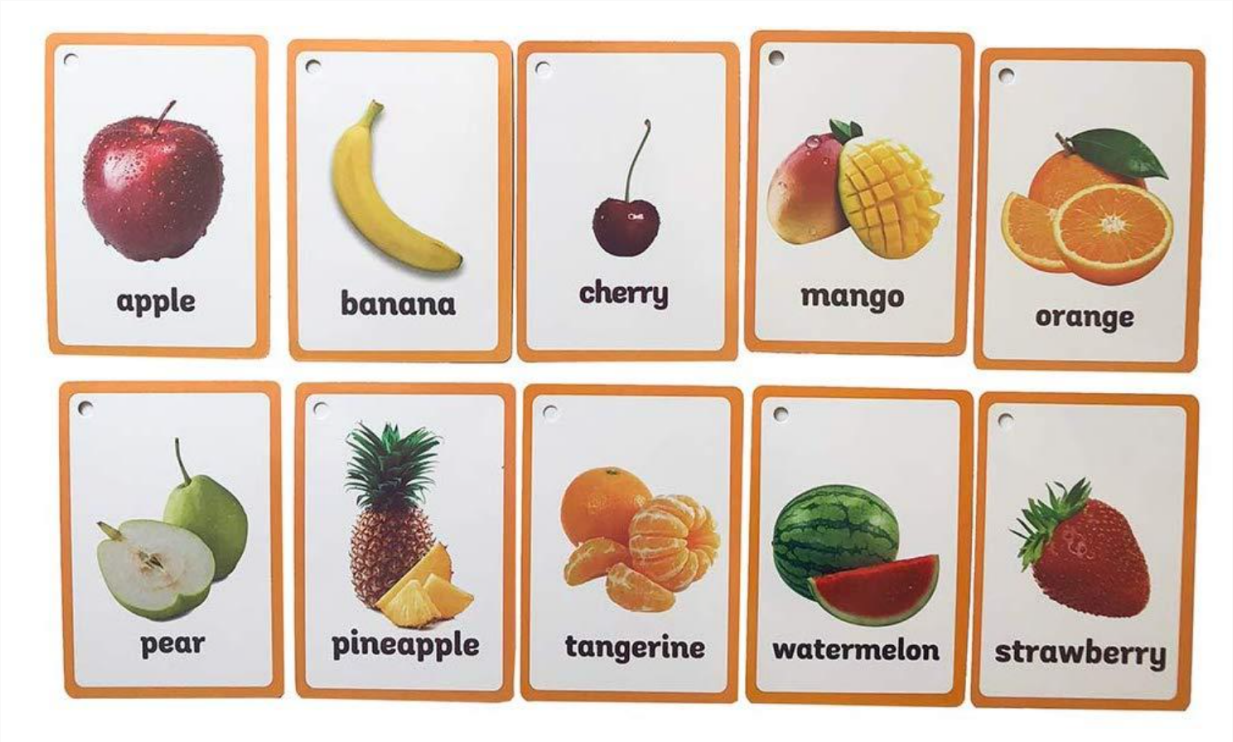
3h ago

Kirsten Luce

Dahmen (2012) "*Photographic Framing in the Stem Cell Debate*"

Why do Images Matter?

People are more likely to recall information learned through visuals



Paivio et al. (1968) "*Why are pictures easier to recall than words?*"

Why do Images Matter?

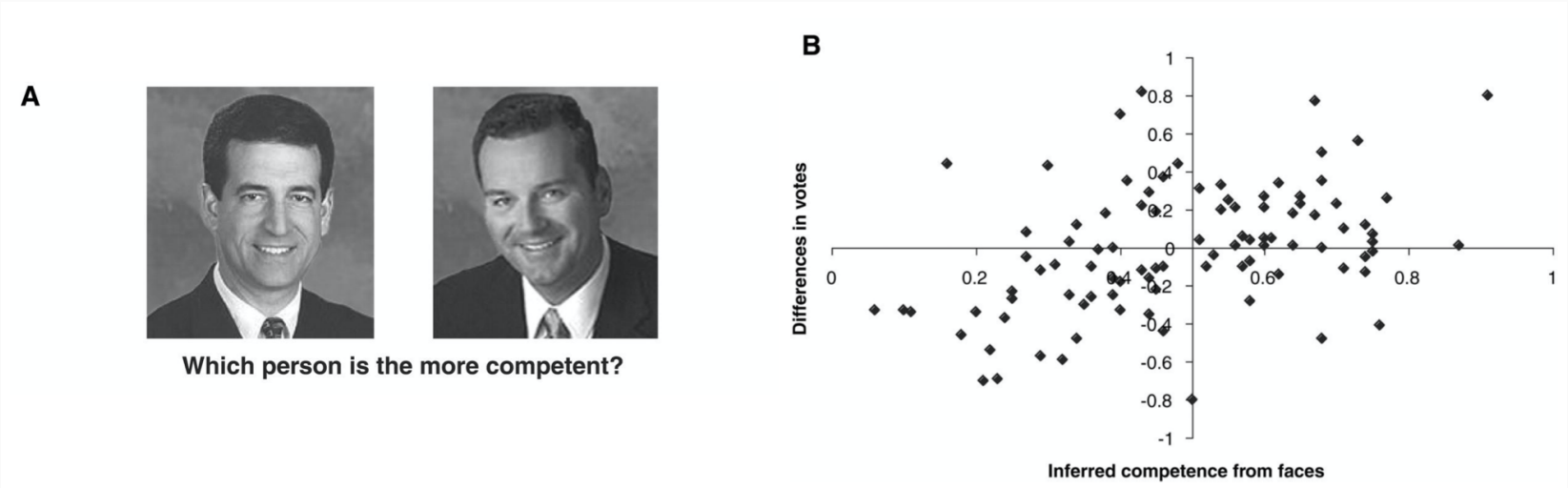
Visuals evoke stronger emotional reactions



Grabe & Bucy (2009) "*Images Bite Politics*"

Why do Images Matter?

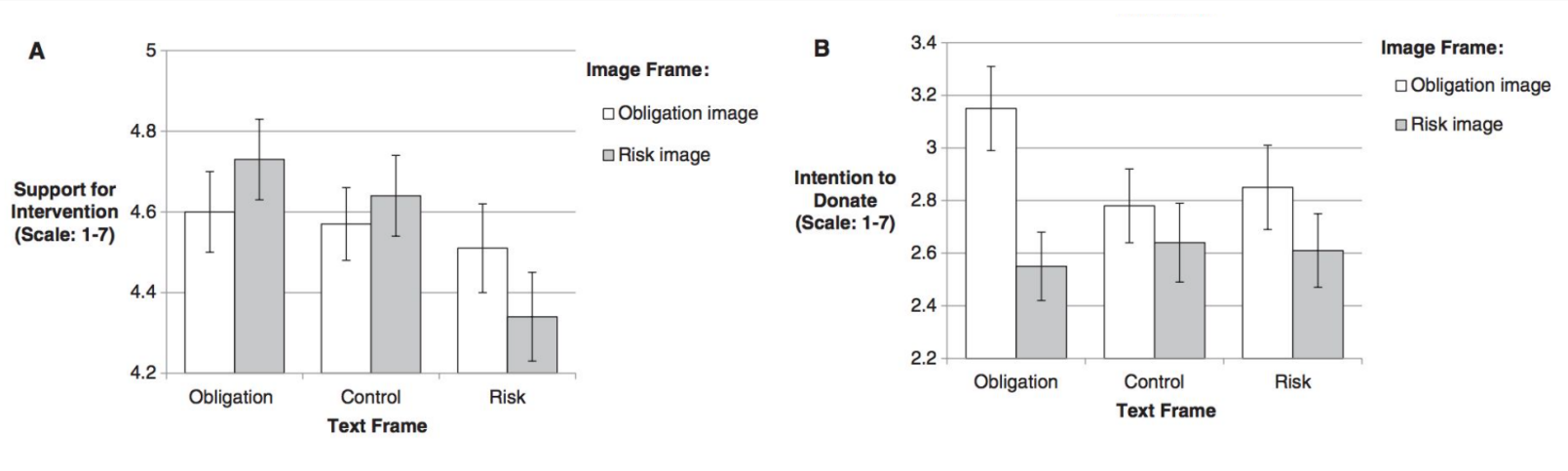
Image effects in **politics**: images $\rightarrow$ inference of competence $\rightarrow$ voting



Todorov et al. (2009) "*Inferences of Competence from Faces Predict Election Outcomes*"

Why do Images Matter?

Image effects in **politics**: images → framing → attitudes



Powell et al. (2015) "A Clearer Picture"

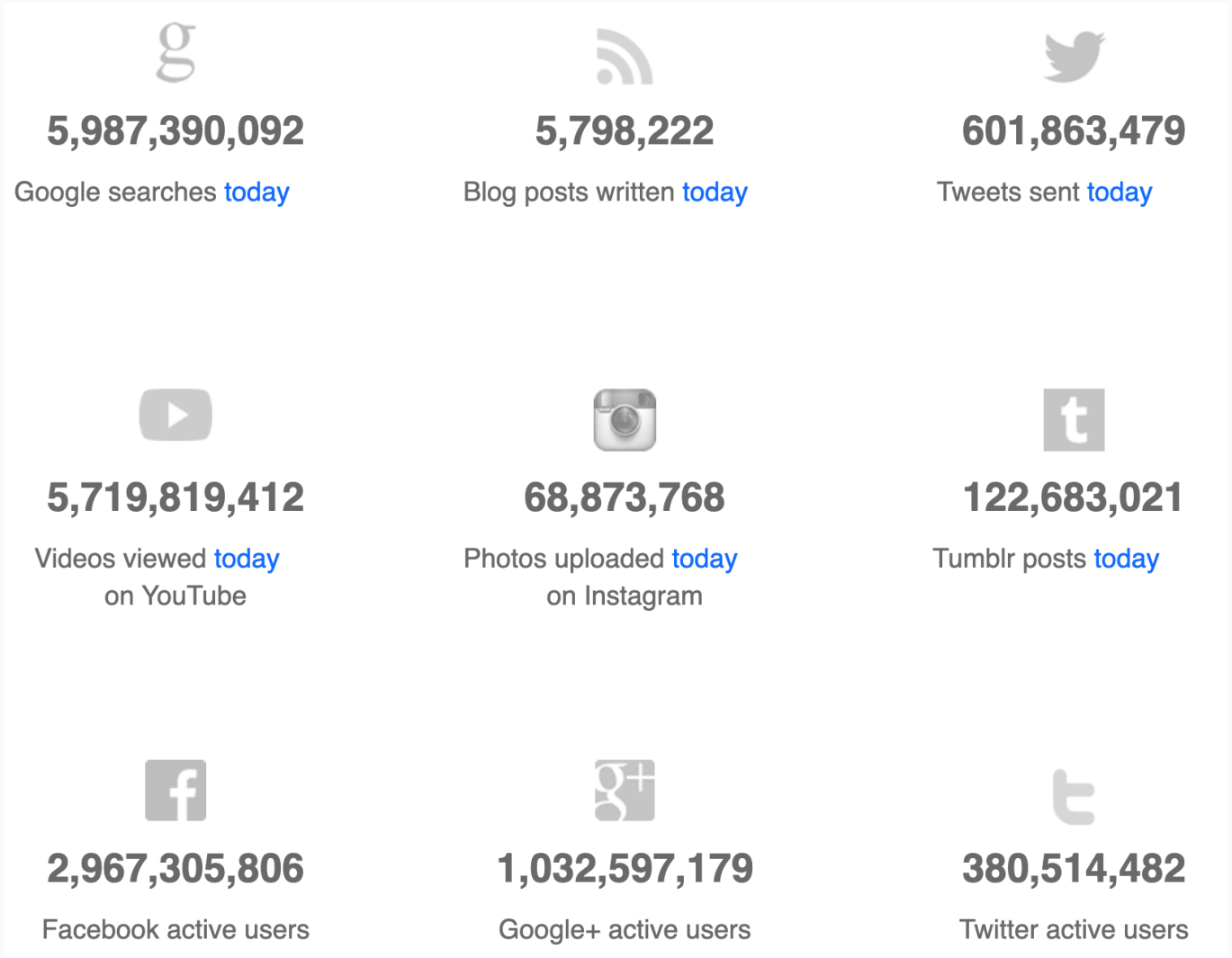
Why do Images Matter?

Image effects in **politics**: images → emotions → **mobilization**

Casas & Webb Williams (2018) "Images That Matter"

Why do Images Matter?

Images are more central than even in our life



Types of Existing Research with Images as Data

Causal Framework

- Images as explanatory/independent variable
  - Boussalis et al. (APSR 2021): *How candidate emotional expressions in televised debate affect voting preferences?*
  - Casas and Webb Williams (PRQ 2018): *Which Black Lives Matter images mobilized more supporters?*
- Images as dependent variable
  - Dietrich and Ko (CCR 2022): *TV coverage of Dr. Fauci during the covid pandemic*
  - Michelle Torres (working paper): *How do different news organizations choose different pictures to accompany articles about Black Lives Matter?*

Types of Existing Research with Images as Data

As a Measurement Strategy

- Images can contain information about electoral incidents and fraud (Callen and Long (2015); Cantú (2019))
- Images can help us identify and classify protest events (Zhang and Pan (2018), Won, Steinert-Threlkeld and Joo (2017))
- Nighttime lights imagery as a proxy for economic development (many authors)
- Digitized historical maps as evidence of road quality variation (Hunziker et al (working paper))
- Videos/Images can help us measure cooperation in legislative politics (Dietrich 2020)

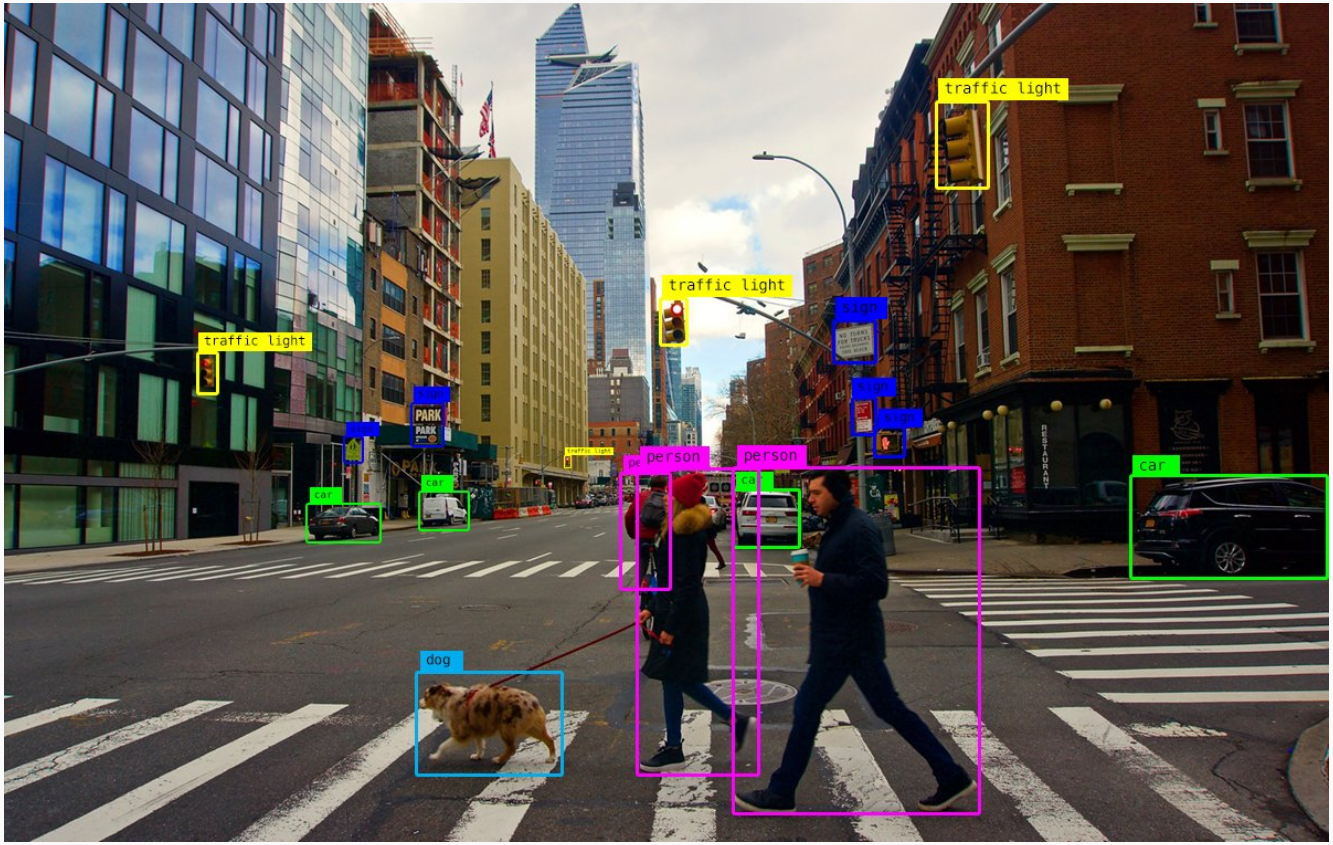
Types of Existing Research with Images as Data

Methodological contributions

- Methodological reviews (Webb Williams et al. 2020; Torres & Cantu 2021)
- Unsupervised clustering (Zhang & Peng 2022; Casas et al.(working paper); Torres (working paper))
- Limitations & biases (Schwemmer et al. 2020)
- Extracting/leveraging aesthetic features (Peng 2021)

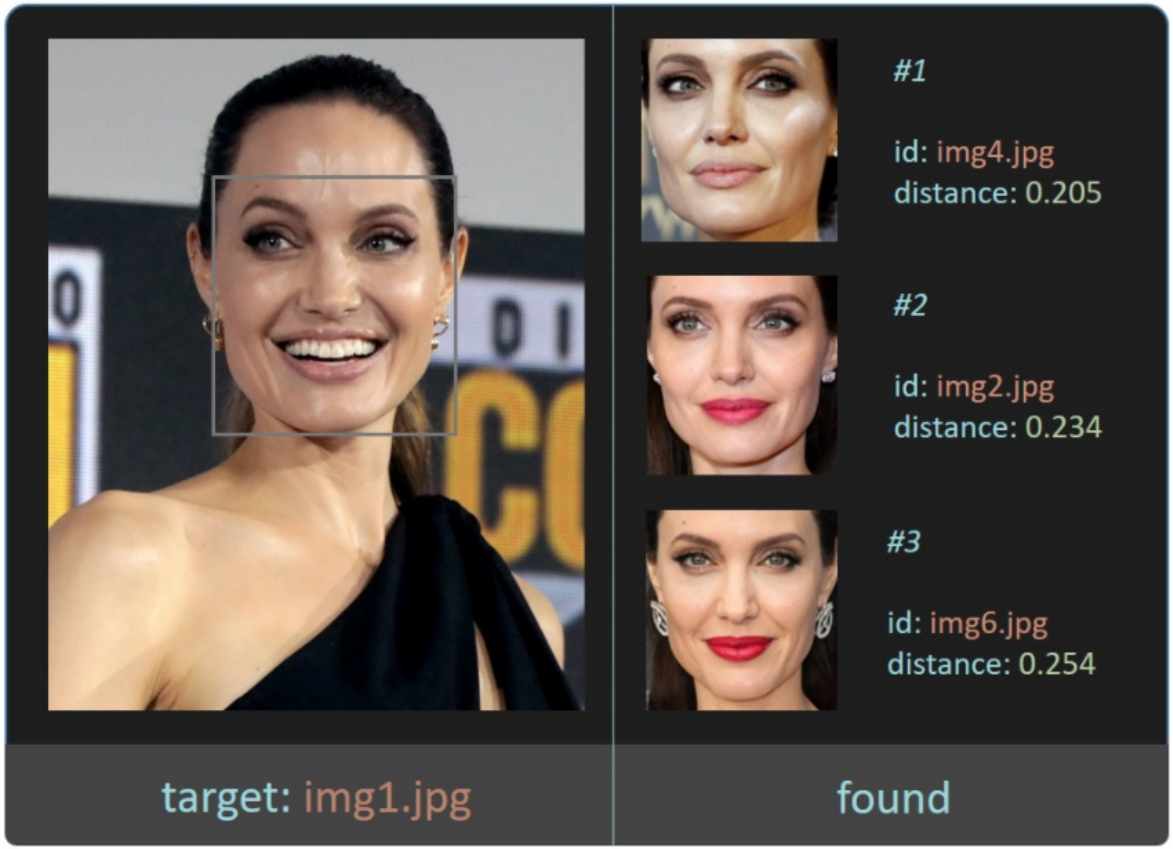
Available Automated Image Analysis Methods

Object detection & recognition



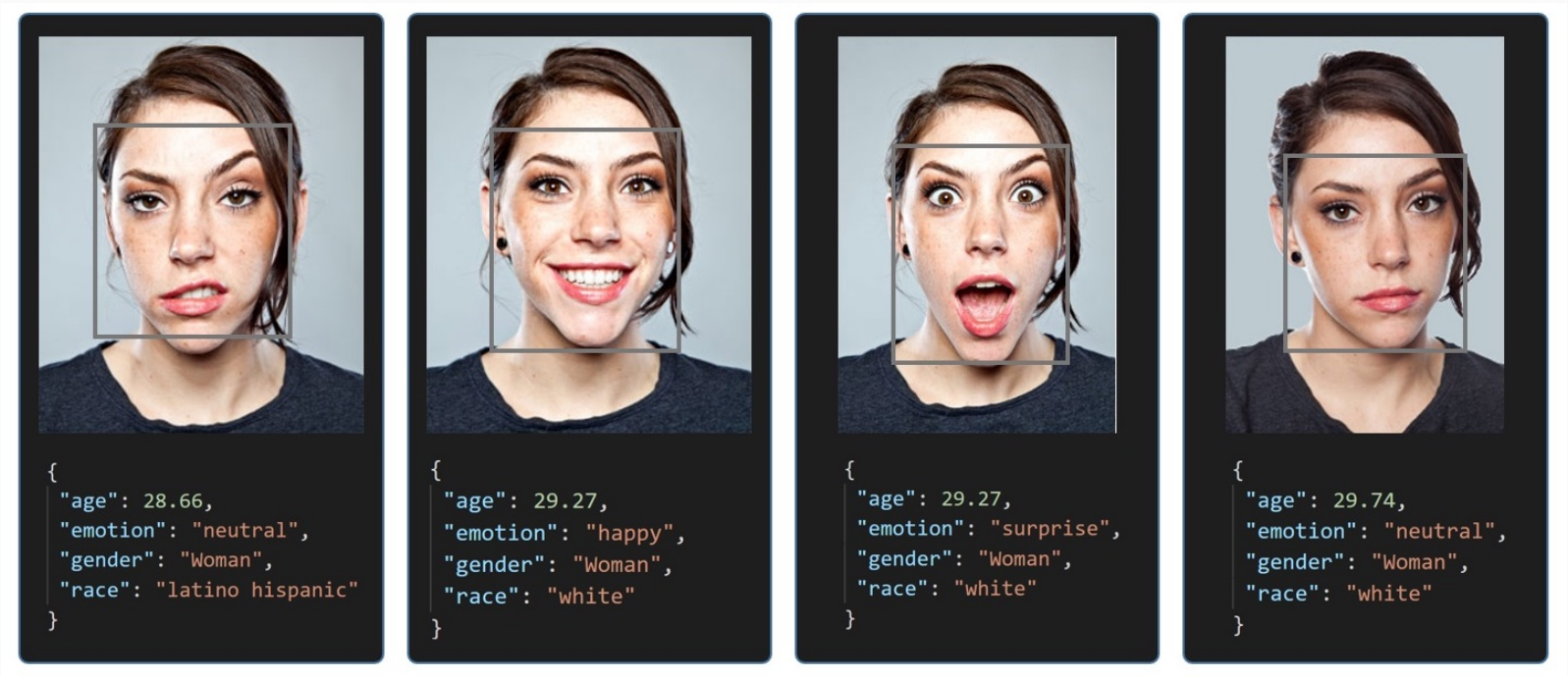
Available Automated Image Analysis Methods

Face detection & recognition



Available Automated Image Analysis Methods

Face analysis



Available Automated Image Analysis Methods

Image Similarity

Available Automated Image Analysis Methods

Unsupervised Clustering

Available Automated Image Analysis Methods

And many others...

- Text extraction (OCR)
- Caption generation
- Sentiment analysis (evoked emotions)
- Visual aesthetics analysis
- etc...

More and more in communication science

A recent turn towards automated content analysis of visual content – including a whole special issue on "images as data" in CCR (Casas & Williams, 2022):

1. Images become more important (e.g., Instagram)
2. More images available to social scientists (not only social media, but also repositories, archives, . . .)
3. New computational methods available

Yet, not as mainstream as compuational textual methods

"Moreover, [. . .] there are starting costs to learning [. . .] computer vision methods, given that the jargon is often very computer-science and machine- learning specific, and state-of-the-art libraries and packages are available mostly in Python while social scientists are often more used to working in the R programming language. Thus the special issue aims to lower start-up costs for scholars interested in using images-as-data methods in their research." (Casas & Williams, 2022, p. 3)

And yet, as we'll see, it's not that big a leap from what we already know!

Some examples

Chen et al., 2022

Can we detect conspiracy videos?

It seems we can, based (also) on visual features like contrast and brightness.

Techniques: Manual feature extraction + classical ML

Some examples

Jürgens et al., 2022

How are age and gender represented on German TV?

There seems to be discrimination against women and elderly

Techniques: Deep learning using pre-trained models

Some examples

Joo and Steinert-Threlkeld, 2022

How do politcians represent themselves visually in the media?

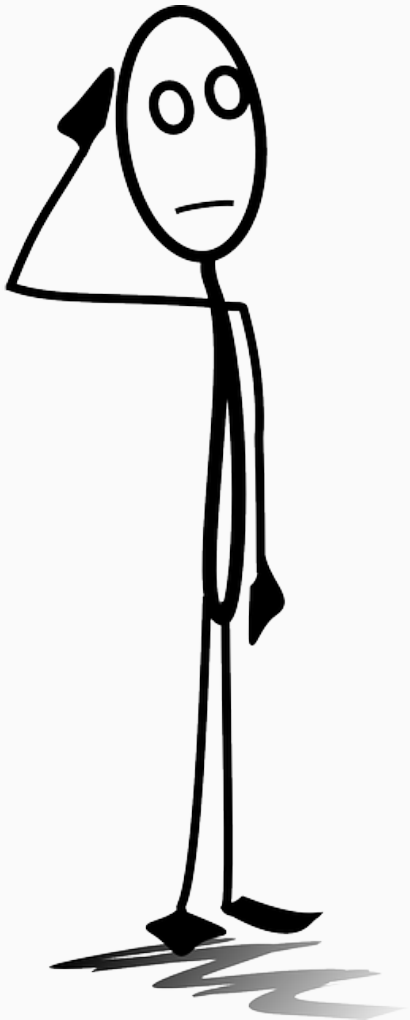
It seems there are party and gender differences

Techniques: Commercial API

You can find some more examples here:
https://computationalcommunication.org/ccr/issue/view/8

Latest trends

- moving images (aka videos)
- combining text and visuals
- really hot: Transformers that model them simultaneously (!!!): "we present data2vec, a framework that uses the same learning method for either speech, NLP or computer vision. The core idea is to predict latent representations of the full input data" (Baevski et al., 2022)



But how do we do all of this? (not data2vec, that's for another time...)

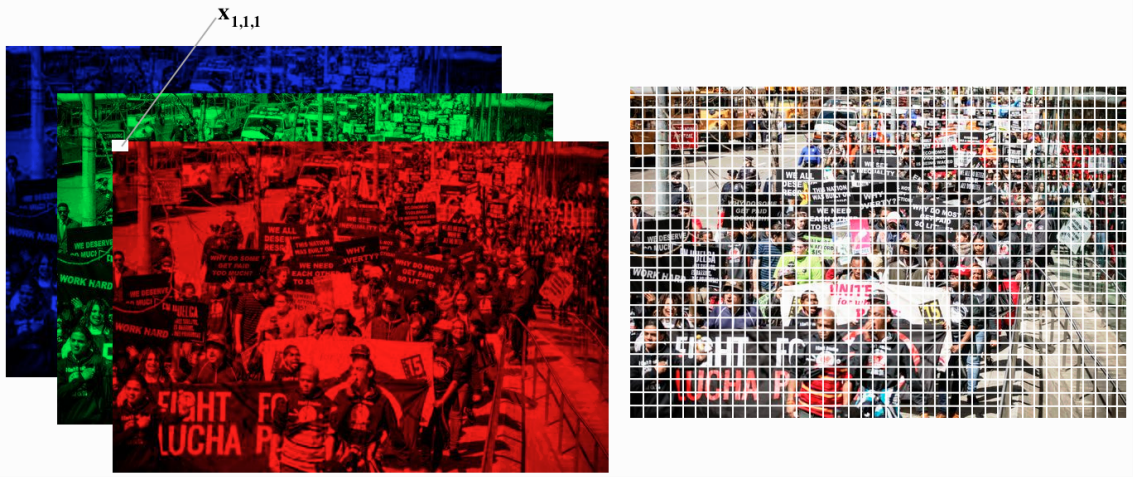
Multimedia data

Representing multimedia data

What exactly is an image to a computer?

- Just a bunch of numbers
- Most images are based on the RGB (red-green-blue) model -> remember primary colors?
- So every image is a three-dimensional matrix with width x height x depth

What an image looks like



$$X = \begin{bmatrix} x_{111} & x_{112} & \dots & x_{11n} \\ x_{121} & x_{122} & \dots & x_{12n} \\ x_{131} & x_{132} & \dots & x_{13n} \\ x_{141} & x_{142} & \dots & x_{14n} \\ \vdots & \vdots & & \\ x_{1n1} & x_{1n2} & \dots & x_{1nn} \end{bmatrix}, \begin{bmatrix} x_{211} & x_{212} & \dots & x_{21n} \\ x_{221} & x_{222} & \dots & x_{22n} \\ x_{231} & x_{232} & \dots & x_{23n} \\ x_{241} & x_{242} & \dots & x_{24n} \\ \vdots & \vdots & & \\ x_{2n1} & x_{2n2} & \dots & x_{2nn} \end{bmatrix}, \begin{bmatrix} x_{311} & x_{312} & \dots & x_{31n} \\ x_{321} & x_{322} & \dots & x_{32n} \\ x_{331} & x_{332} & \dots & x_{33n} \\ x_{341} & x_{342} & \dots & x_{34n} \\ \vdots & \vdots & & \\ x_{3n1} & x_{3n2} & \dots & x_{3nn} \end{bmatrix}$$

Amount of pixels

- A Grayscale image only has one layer, so for a 200×300 image, we have a two-dimensional matrix with $200 \times 300 = 60,000$ pixels
- The same size image in color is: $200 \times 300 \times 3 = 180,000$ integers between 0 and 255

What can we do with images in Python?

- PIL (Pillow), the Python Image Library, provides many typical image operations (reading, displaying, cropping, color transformations)
- That's cool for batch processing (a for-loop and you can resize thousands of images...)
- But more imporantly: **It's just a vector/matrix of integers, so we can do machine learning just like we did before!**

Multimedia data

Classic SML

First approach

- If each picture is just a vector of integers, scikit-learn works *exactly the same* as for survey data (Chapter 8) or text (Chapter 11).
- Typical example: recognizing hand-written characters with a Random Forest (Example 14.10) or a Support Vector Machine[1]

[1] https://scikit-learn.org/stable/auto_examples/classification/plot_digits_classification.html

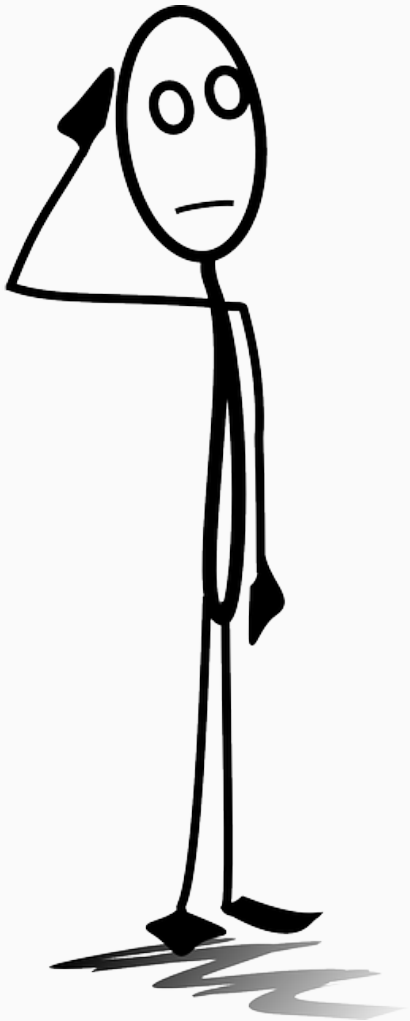
This is impressive!

```
Classification report for classifier SVC(gamma=0.001):
              precision    recall  f1-score   support

           0       1.00      0.99      0.99        88
           1       0.99      0.97      0.98        91
           2       0.99      0.99      0.99        86
           3       0.98      0.87      0.92        91
           4       0.99      0.96      0.97        92
           5       0.95      0.97      0.96        91
           6       0.99      0.99      0.99        91
           7       0.96      0.99      0.97        89
           8       0.94      1.00      0.97        88
           9       0.93      0.98      0.95        92

    accuracy                           0.97       899
   macro avg       0.97      0.97      0.97       899
weighted avg       0.97      0.97      0.97       899
```

https://scikit-learn.org/stable/auto_examples/classification/plot_digits_classification.html

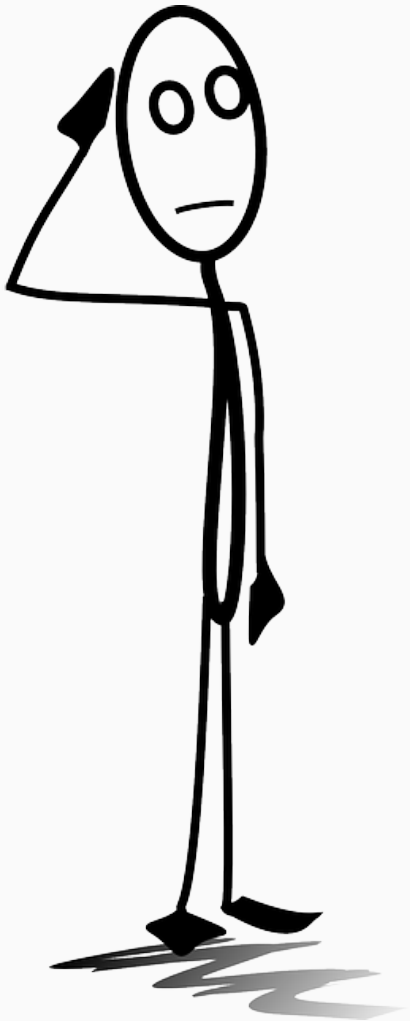


Can you explain why this works?

The intuition behind it

If the images are cropped and resized equally (!). . .

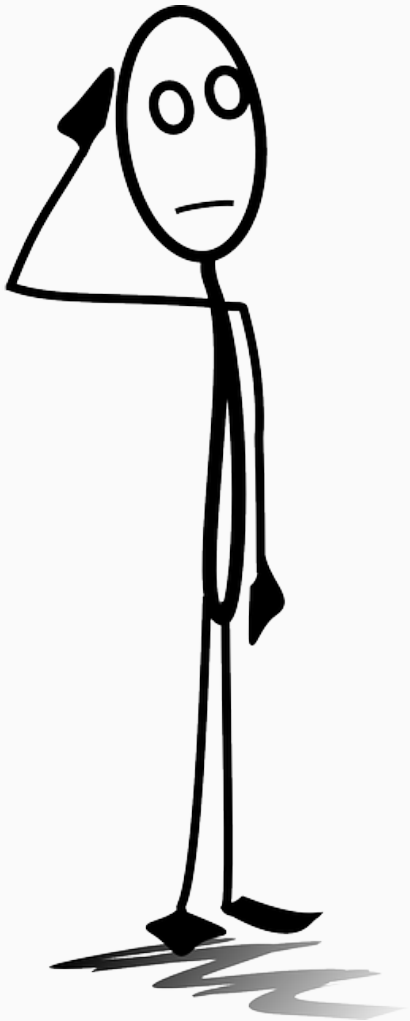
. . . then the pixels in the center should be white for a 0 but dark for a 8.



Can you name tasks for which you think this would not work?

Multimedia data

Deep Learning



Do you remember what the reason for using deep learning in text classification was?

Say we want to recognize whether an image is a cat or a dog...

- Can we really say that the color of a pixel maps directly to the "catness" or "dogness" of the picture?
- Or should it not rather be about their fur, the shape of the ears, ...
- But it seems impossible to engineer these features by hand :-(

Let's use a deep neural network to learn them!

Say we want to recognize whether an image is a cat or a dog...

- Can we really say that the color of a pixel maps directly to the "catness" or "dogness" of the picture?
- Or should it not rather be about their fur, the shape of the ears, ...
- But it seems impossible to engineer these features by hand :-(

Let's use a deep neural network to learn them!

How do we do this?

- For deep learning, we use `keras` instead of `scikit-learn`
- You need to specify the *architecture* of the network
- The process is resource-intensive

Working with images

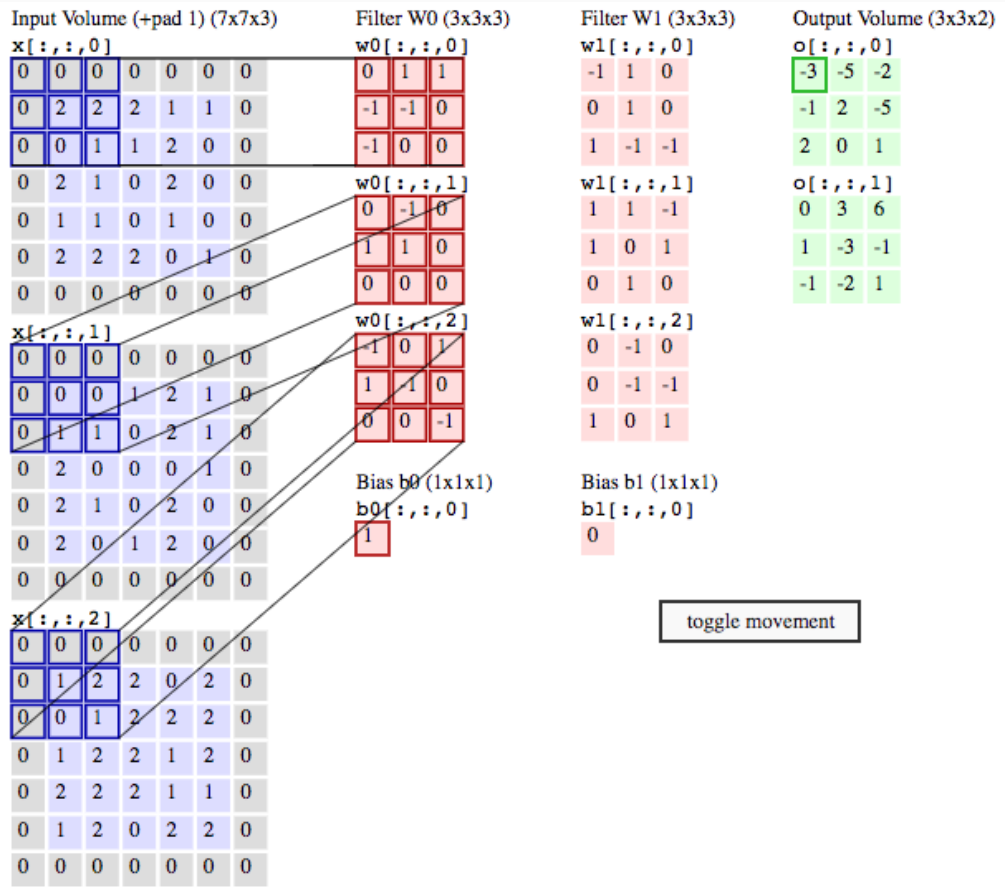
There are many frameworks, but particularly interesting for us:

- The Convolutional Neural Network (CNN): A neural network in which the layers are not fully connected
- We have talked about this before, remember?

Convolutional Neural Nets for Computer Vision

Convolutional layers: weights (**filters**) are not connected to the whole **input volume**: convolution.

Click here for a full visualization by the Stanford cs231 folks.



Some terminology

- **input volume**: a 3-dimensional input
- **convolutional layer**: a 4-dimensional parameter layer where convolutional filters are applied to the input volume; of size FxFxNxK where F is the width and height of the filter, N is the number of filter dimensions, and K is the number of filters $\rightarrow$ 3x3x3x2 in the previous example
- **stride**: the number of pixels we move the filter at a time. This is 2 in the previous example
- **zero-padding**: adding zeros around the input border (often done to avoid deforming input images)
- **pooling layer**: a layer where we reduce the size the output of a convolutional layer. From 224x224x3x64 to 112x112x3x64 for example

Some terminology

- **input volume**: a 3-dimensional input
- **convolutional layer**: a 4-dimensional parameter layer where convolutional filters are applied to the input volume; of size FxFxNxK where F is the width and height of the filter, N is the number of filter dimensions, and K is the number of filters $\rightarrow$ 3x3x3x2 in the previous example
- **stride**: the number of pixels we move the filter at a time. This is 2 in the previous example
- **zero-padding**: adding zeros around the input border (often done to avoid deforming input images)
- **pooling layer**: a layer where we reduce the size the output of a convolutional layer. From 224x224x3x64 to 112x112x3x64 for example

Some terminology

- **input volume**: a 3-dimensional input
- **convolutional layer**: a 4-dimensional parameter layer where convolutional filters are applied to the input volume; of size FxFxNxK where F is the width and height of the filter, N is the number of filter dimensions, and K is the number of filters $\rightarrow$ 3x3x3x2 in the previous example
- **stride**: the number of pixels we move the filter at a time. This is 2 in the previous example
- **zero-padding**: adding zeros around the input border (often done to avoid deforming input images)
- **pooling layer**: a layer where we reduce the size the output of a convolutional layer. From 224x224x3x64 to 112x112x3x64 for example

Some terminology

- **input volume**: a 3-dimensional input
- **convolutional layer**: a 4-dimensional parameter layer where convolutional filters are applied to the input volume; of size FxFxNxK where F is the width and height of the filter, N is the number of filter dimensions, and K is the number of filters $\rightarrow$ 3x3x3x2 in the previous example
- **stride**: the number of pixels we move the filter at a time. This is 2 in the previous example
- **zero-padding**: adding zeros around the input border (often done to avoid deforming input images)
- **pooling layer**: a layer where we reduce the size the output of a convolutional layer. From 224x224x3x64 to 112x112x3x64 for example

Some terminology

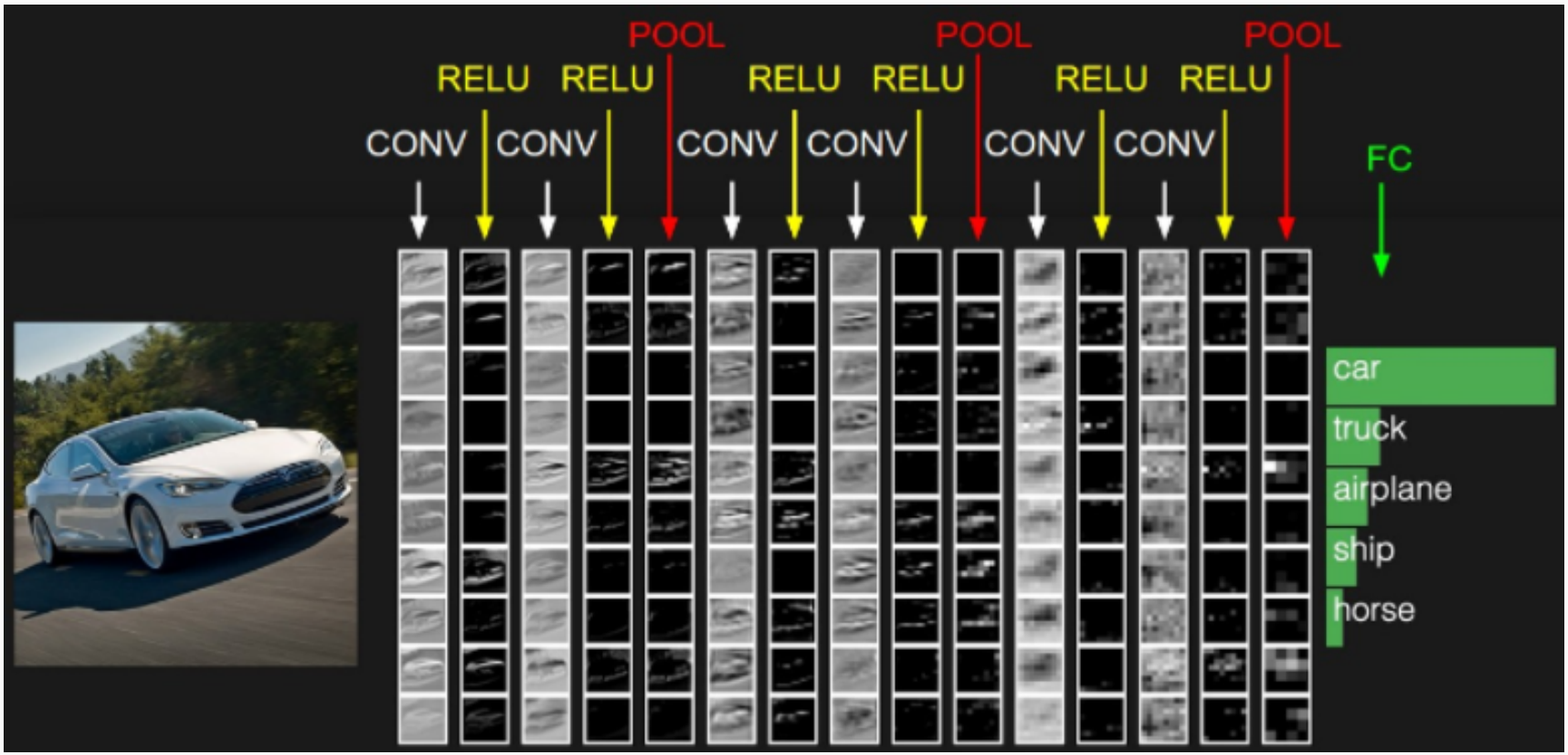
- **input volume**: a 3-dimensional input
- **convolutional layer**: a 4-dimensional parameter layer where convolutional filters are applied to the input volume; of size FxFxNxK where F is the width and height of the filter, N is the number of filter dimensions, and K is the number of filters $\rightarrow$ 3x3x3x2 in the previous example
- **stride**: the number of pixels we move the filter at a time. This is 2 in the previous example
- **zero-padding**: adding zeros around the input border (often done to avoid deforming input images)
- **pooling layer**: a layer where we reduce the size the output of a convolutional layer. From 224x224x3x64 to 112x112x3x64 for example

...and more terminology

- **fully connected layer**: a layer of weights that is connected to the whole input volume. These are usually at the end of a network.
- **softmax**: a multi-class classifier. This is basically a multinomial logit model that uses the output of the last fully-connected layer to predict the final classes of interest

Convolutional Neural Nets for Computer Vision

This is how a ConvNet looks like



Training your own neural network?

If you have a very specific task and an annotated dataset – sure!

But:

- Can be *very* resource-intensive
- Often, very large amounts of training data needed
- There are so many architectures, and you can't possibly know the best ones without being an expert

Training your own neural network?

Solution 1: Using a pre-trained model

Just use a CNN that already is trained (Examples 14.14–14.17).

Make sure to read up on the specific model you are using.

Training your own neural network?

Solution 2: Fine-tuning a pre-trained model

If you have a labeled dataset, you can start with an existing pre-trained model and then fine-tune it with your data (many tutorials online)

There is also a really, really cool approach called CLIP that uses transformers to embed images and texts in the same (!) vector space (Radford et al., 2021) –
https://huggingface.co/docs/transformers/model_doc/clip

Training your own neural network?

Solution 3: A (commercial) API

(next section)

Multimedia data

(Commercial) APIs

What is it?

- You send an image to the API and get a set of labels back
- Under the hood: pre-trained neural networks
- Very much black box: you have no idea where the labels come from
- Prominent players: Clarifai, Google Cloud Vision, Microsoft
- Usually paid but often free for small projects and/or academic use

Automated Visual Content Analysis (Araujo et al., 2020)

Problem

The labels do not correspond what one may be theoretically interested in.

Solution

- Annotate a set of images manually using a codebook
- Use the labels provided by the API as features to train a classifier to predict the annotations
- Thus, we essentially have transformed the image prediction task to a textual prediction task

Automated Visual Content Analysis (Araujo et al., 2020)

Problem

The labels do not correspond what one may be theoretically interested in.

Solution

- Annotate a set of images manually using a codebook
- Use the labels provided by the API as features to train a classifier to predict the annotations
- Thus, we essentially have transformed the image prediction task to a textual prediction task

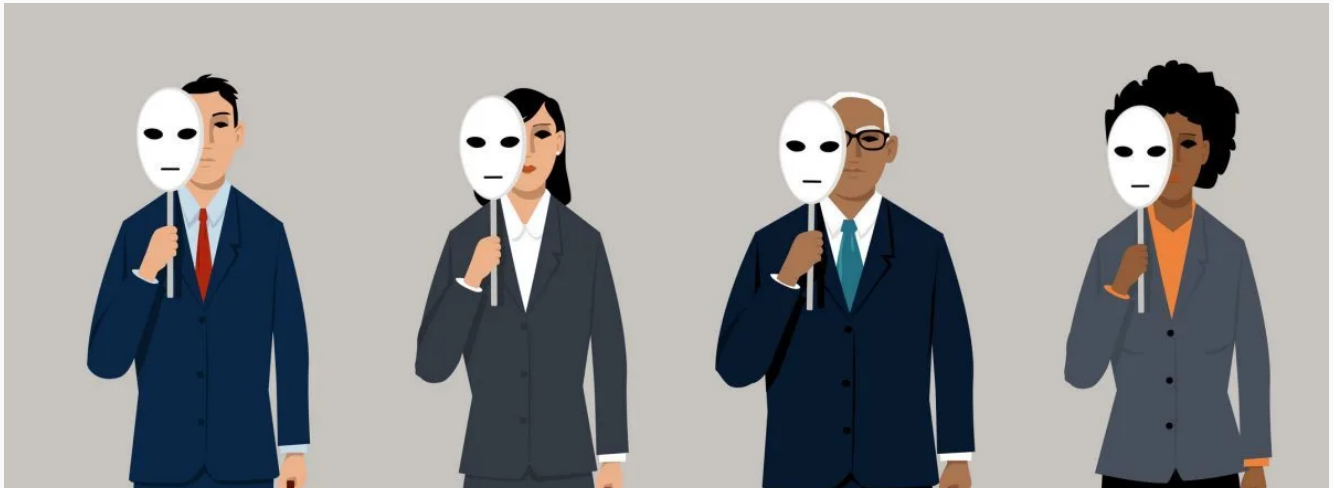
Is it difficult to use a computer vision API?

No.

All providers provide Python code examples, e.g. https://cloud.google.com/vision/docs/detect-labels-image-client-libraries

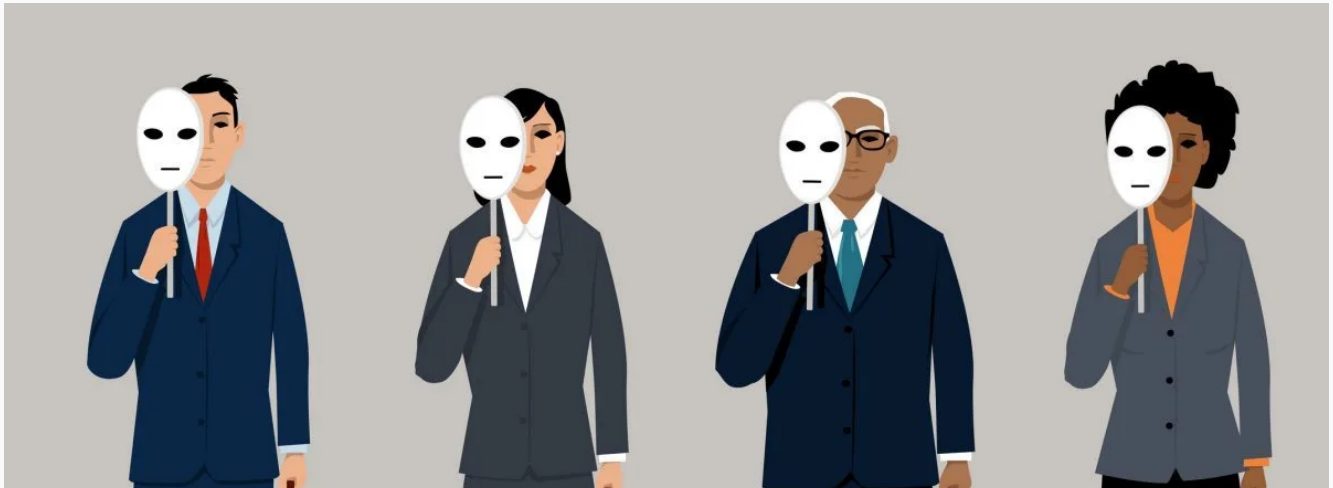
Biases: Garbage in, garbage out

- Known issues: Gender and Race
- Joy Buolamwini (MIT) → Gender Shades
- Fairness, Accountability, and Transparency conference
- Best option: Knowing the training data, proper validation
- Schwemmer et al (2020) show an example on how to detect biases in commercial models



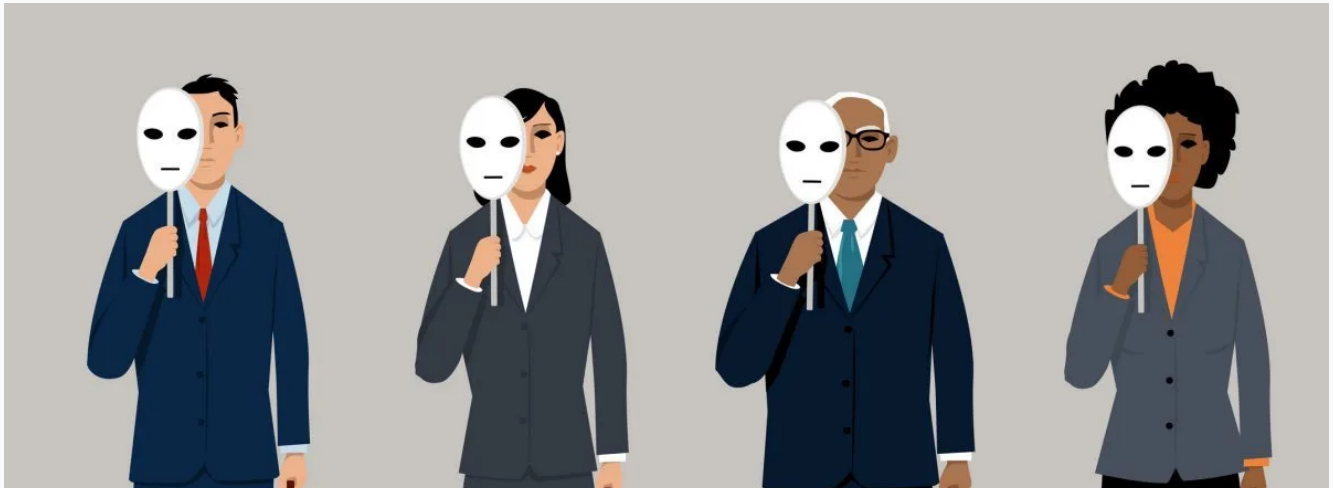
Biases: Garbage in, garbage out

- Known issues: Gender and Race
- Joy Buolamwini (MIT) $\rightarrow$ Gender Shades
- Fairness, Accountability, and Transparency conference
- Best option: Knowing the training data, proper validation
- Schwemmer et al (2020) show an example on how to detect biases in commercial models



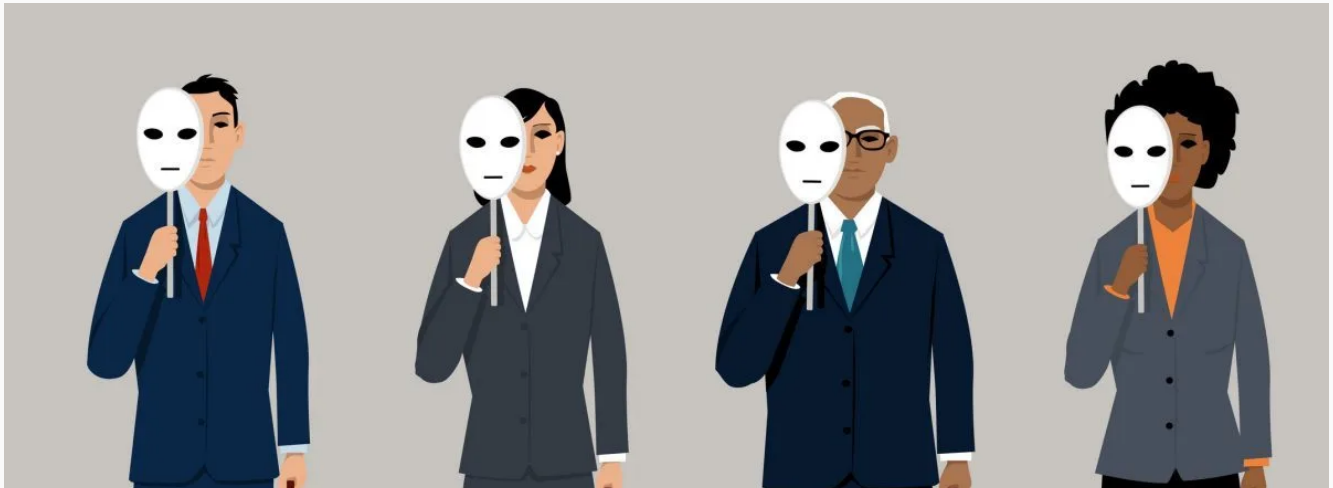
Biases: Garbage in, garbage out

- Known issues: Gender and Race
- Joy Buolamwini (MIT) $\rightarrow$ Gender Shades
- Fairness, Accountability, and Transparency conference
- Best option: Knowing the training data, proper validation
- Schwemmer et al (2020) show an example on how to detect biases in commercial models



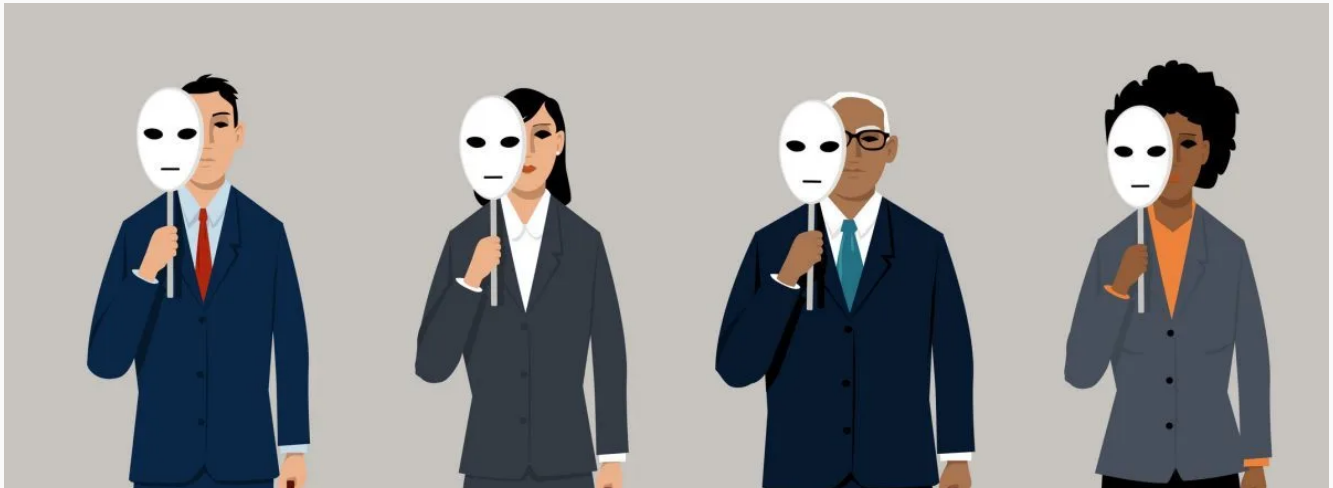
Biases: Garbage in, garbage out

- Known issues: Gender and Race
- Joy Buolamwini (MIT) $\rightarrow$ Gender Shades
- Fairness, Accountability, and Transparency conference
- Best option: Knowing the training data, proper validation
- Schwemmer et al (2020) show an example on how to detect biases in commercial models



Biases: Garbage in, garbage out

- Known issues: Gender and Race
- Joy Buolamwini (MIT) $\rightarrow$ Gender Shades
- Fairness, Accountability, and Transparency conference
- Best option: Knowing the training data, proper validation
- Schwemmer et al (2020) show an example on how to detect biases in commercial models

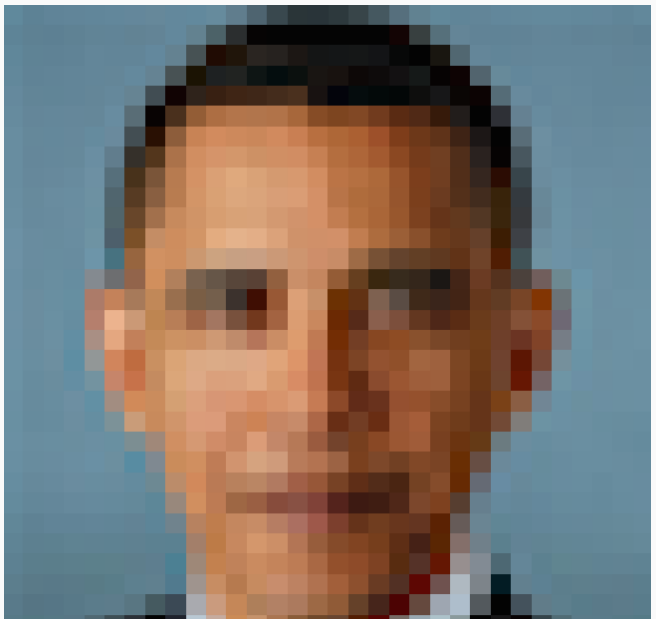


A recent example

Dutch student files complaint with the Netherlands Institute for Human Rights about the use of racist software by her university

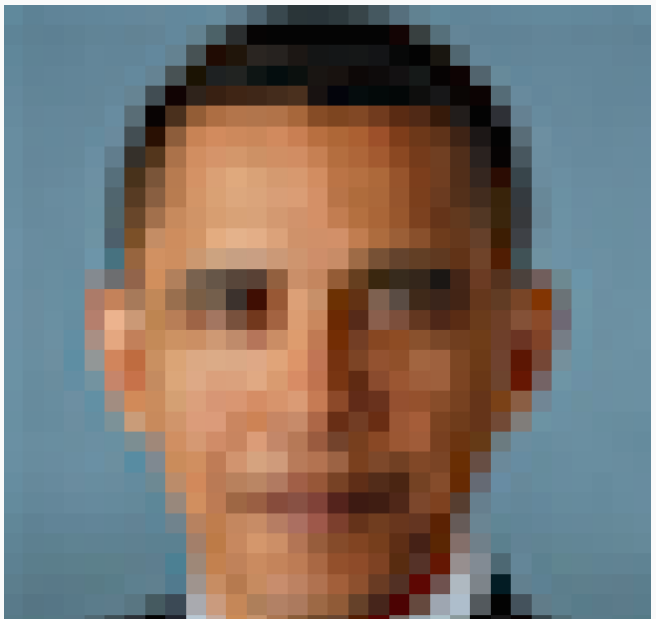
Privacy: My face is on that picture

- GDPR: Private data needs to be anonymized
- Especially important when working with social media images
- Common strategies: Pixellation & Blurring (not bulletproof!)
- Sharing of data? Using external services for data? $\rightarrow$ Consider using open-source taggers run locally



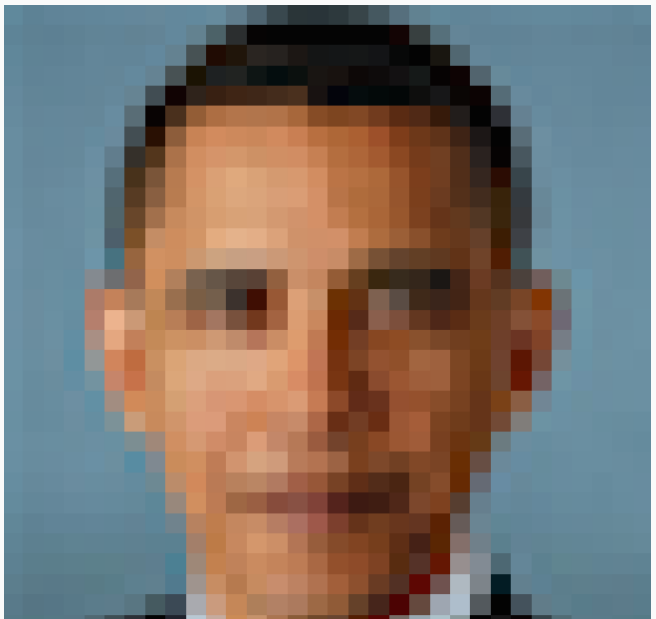
Privacy: My face is on that picture

- GDPR: Private data needs to be anonymized
- Especially important when working with social media images
- Common strategies: Pixellation & Blurring (not bulletproof!)
- Sharing of data? Using external services for data? → Consider using open-source taggers run locally



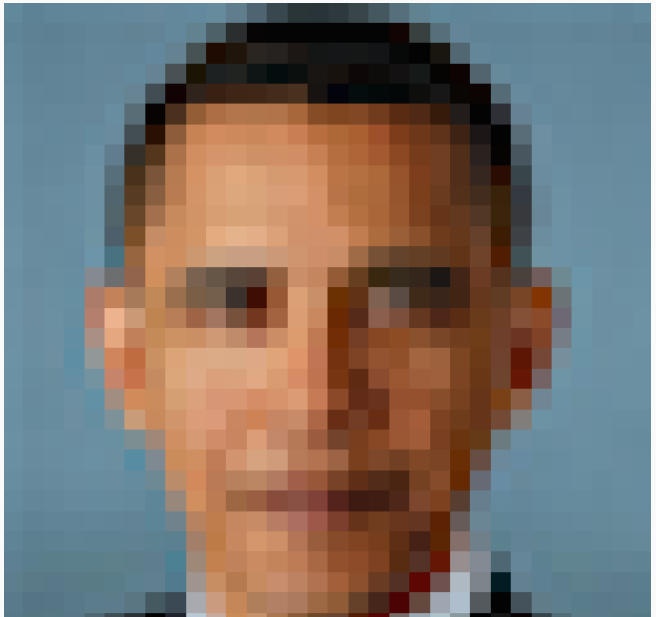
Privacy: My face is on that picture

- GDPR: Private data needs to be anonymized
- Especially important when working with social media images
- Common strategies: Pixellation & Blurring (not bulletproof!)
- Sharing of data? Using external services for data? → Consider using open-source taggers run locally

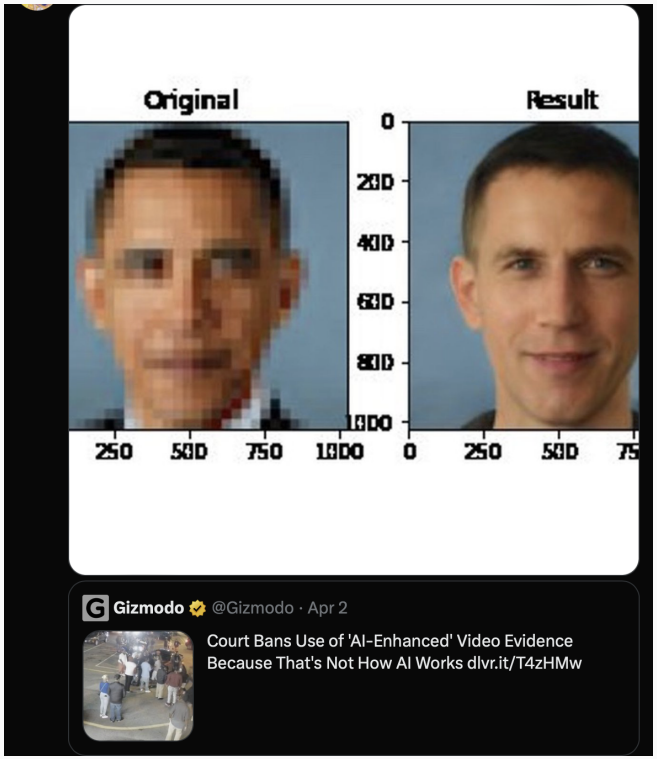


Privacy: My face is on that picture

- GDPR: Private data needs to be anonymized
- Especially important when working with social media images
- Common strategies: Pixellation & Blurring (not bulletproof!)
- Sharing of data? Using external services for data? $\rightarrow$ Consider using open-source taggers run locally



Privacy: My face is on that picture



Further reading

- all the articles in the special issue edited by Casas and Williams, 2022
- a short book by Webb Williams et al., 2020
- the study by Araujo et al., 2020
- This medium post on CNNs: https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939

Final remark

The black box nature of (most of) these techniques *is* a big problem from a social-science and epistemological point of view: in particular, known and un-known biases!

As always, but maybe especially here: Never take the outcome as a given – it needs interpretation and critical reflection!

Next steps

Let's have a look together at the course manual regarding the final project.

Friday

On Friday, you can *either* work on one of the computer vision examples discussed today and/or in the book.

Or you start working on your final project.

References

Araujo, T., Lock, I., & van de Velde, B. (2020). **Automated visual content analysis (AVCA) in communication research: A protocol for large scale image classification with pre-trained computer vision models.** *Communication Methods and Measures*, *14*(4), 239–265. https://doi.org/10.1080/19312458.2020.1810648

Baevski, A., Hsu, W.-N., Xu, Q., Babu, A., Gu, J., & Auli, M. (2022). **data2vec: A General Framework for Self-supervised Learning in Speech, Vision and Language.** http://arxiv.org/abs/2202.03555

Casas, A., & Williams, N. W. (2022). **Introduction to the special issue on images as data.** *Computational Communication Research*, *4*(1). https://doi.org/10.5117/ccr2022.1.000.casa


Chen, K., Kim, S. J., Gao, Q., & Raschka, S. (2022). **Visual framing of science conspiracy videos.** *Computational Communication Research*, *4*(1), 98–134. https://doi.org/10.5117/ccr2022.1.003.chen


Joo, J., & Steinert-Threlkeld, Z. C. (2022). **Image as data: Automated content analysis for visual presentations of political actors and events.** *Computational Communication Research*, *4*(1), 11–67. https://doi.org/10.5117/ccr2022.1.001.joo


Jürgens, P., Meltzer, C. E., & Scharkow, M. (2022). **Visual framing of science conspiracy videos.** *Computational Communication Research*, *4*(1), 173–207. https://doi.org/10.5117/ccr2022.1.005.jurg

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). **Learning transferable visual models from natural language supervision.** In M. Meila & T. Zhang (Eds.), *Proceedings of the 38th international conference on machine learning* (pp. 8748–8763, Vol. 139). PMLR. https://proceedings.mlr.press/v139/radford21a.html

Webb Williams, N., Casas, A., & Wilkerson, J. D. (2020). ***Images as data for social science research: An introduction to convolutional neural nets for image classification*.** Cambridge University Press. https://doi.org/10.1017/9781108860741