CS 1674: Intro to Computer Vision

Introduction

Prof. Adriana Kovashka
University of Pittsburgh
August 28, 2023
Course Info

• **Course website:** [http://people.cs.pitt.edu/~kovashka/cs1674_fa23](http://people.cs.pitt.edu/~kovashka/cs1674_fa23)

• **Instructor:** Adriana Kovashka  
  ([kovashka@cs.pitt.edu](mailto:kovashka@cs.pitt.edu))

• **Office:** Sennott Square 5325

• **Class:** Mon/Wed, 1pm-2:15pm

• **Office hours:** Mon/Wed, 11am-1pm
About the Instructor

Born 1985 in Sofia, Bulgaria

Got BA in 2008 at Pomona College, CA (Computer Science & Media Studies)

Got PhD in 2014 at University of Texas at Austin (Computer Vision)
About the TA (TBD)

- **TA**: Mesut Erhan Unal
- **Office**: SenSq 5404
- **Office hours**: TBD

  - Do this Doodle by the end of Friday:
    [https://www.when2meet.com/?20648101-cIWNYY](https://www.when2meet.com/?20648101-cIWNYY)

Mark *all* times you’re available, in a regular week.
Course Goals

• To learn the basics of low-level image analysis
• To learn about some classic and modern approaches to high-level computer vision tasks (e.g. object recognition)
• To get experience with vision techniques
• To learn/apply basic machine learning (a key component of modern computer vision)
• To think critically about vision approaches, and to see connections between works
Textbooks

- **Computer Vision: Algorithms and Applications** by Richard Szeliski, 2nd edition
- **Visual Object Recognition** by Kristen Grauman and Bastian Leibe
- More resources available on course webpage

- Your notes from class are your best study material, slides are *not* complete with notes
Programming Language

• We’ll use Matlab
• It can be downloaded for free from MyPitt -> Software Downloads
• Please download latest version
• We’ll do a short tutorial; ask TA if you need further help
Course Structure (CS 1674)

- Lectures
- Homework
  - 9 assignments x 6% each = 54%
- Exams
  - First exam (20%)
  - Second exam (20%)
- Participation (6%)
Course Structure (CS 2074)

- Lectures
- Homework
  - 9 assignments x 5% each = 45%
- Exams
  - First exam (15%)
  - Second exam (15%)
- Participation (5%)
- Course project
  - Proposal (3%)
  - Status report (5%)
  - Presentation (5%)
  - Final report (7%)
Policies and Schedule

http://people.cs.pitt.edu/~kovashka/cs1674_fa23
Warning #1

• This class is **a lot of work**

• I’ve opted for shorter, more manageable HW assignments, but there is a lot of them

• I expect you’d be spending **6-8 hours** on homework each week

• ... But you get to understand algorithms and concepts in detail!
Warning #2

• Some parts will be **hard** and require that you pay close attention!

• **Use instructor’s and TA’s office hours**

• ... You will learn a lot!
Questions?
Plan for Today

• Blitz introductions
• What is computer vision?
  – Why do we care?
  – What are the challenges?
  – What is recent research like?
• Overview of topics
• Review and tutorial
  – Linear algebra
  – Matlab
Blitz introductions (15 sec)

• What is your name?
• What one thing outside of school are you passionate about?
• What do you hope to get out of this class? (optional)

• Every time you speak, please remind me your name
Computer Vision
What is computer vision?

Done?

"We see with our brains, not with our eyes“  (Oliver Sacks and others)
What is computer vision?

• Automatic understanding of images and video
  – Algorithms and representations to allow a machine to recognize objects, people, scenes, and activities
  – Algorithms to mine, search, and interact with visual data
  – Computing properties and navigating within the 3D world using visual data
  – Generating realistic synthetic visual data
Perception and interpretation

How to do? What are the challenges?
Visual search, organization

Query → Image or video archives → Relevant content

Kristen Grauman
Measurement

Real-time stereo

Structure from motion

Multi-view stereo for community photo collections

Pollefeys et al.

Goesele et al.
Generation

Karras et al., “Progressive Growing of GANs for Improved Quality, Stability, and Variation”, ICLR 2018
Related disciplines

- Artificial intelligence
- Machine learning
- Cognitive science
- Graphics
- Image processing
- Algorithms

Kristen Grauman
Inverse problems: analysis and synthesis.
Why vision?

• Images and video are everywhere!

144k hours uploaded to YouTube daily
4.5 mil photos uploaded to Flickr daily
10 bil images indexed by Google

- Personal photo albums
- Movies, news, sports
- Surveillance and security
- Medical and scientific images

Adapted from Lana Lazebnik
Why vision?

• As image sources multiply, so do applications
  – Relieve humans of boring, easy tasks
  – Perception for robotics / autonomous agents
  – Organize and give access to visual content
  – Description of content for the visually impaired
  – Human-computer interaction
  – Fun applications (e.g. art styles to my photos)
  – What else?

Adapted from Kristen Grauman
Computer vision tools already used in practice
Faces and digital cameras

Camera waits for everyone to smile to take a photo [Canon]

Setting camera focus via face detection
Optical character recognition

INPUT: IMAGE

OUTPUT: TEXT

MATLAB OCR

Accurate object detection in real time

<table>
<thead>
<tr>
<th></th>
<th>Pascal 2007 mAP</th>
<th>Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPM v5</td>
<td>33.7</td>
<td>.07 FPS</td>
</tr>
<tr>
<td>R-CNN</td>
<td>66.0</td>
<td>.05 FPS</td>
</tr>
<tr>
<td>Fast R-CNN</td>
<td>70.0</td>
<td>.5 FPS</td>
</tr>
<tr>
<td>Faster R-CNN</td>
<td>73.2</td>
<td>7 FPS</td>
</tr>
<tr>
<td>YOLO</td>
<td>69.0</td>
<td>45 FPS</td>
</tr>
</tbody>
</table>

Exploring photo collections

Photo Tourism
Exploring photo collections in 3D

Snavely et al.

Kristen Grauman
Linking to info with a mobile device

Situated search
Yeh et al., MIT

MSR Lincoln

kooaba
Transferring art styles

DeepArt.io – try it for yourself!

Gatys et al., CVPR 2016
Image generation

Choi et al., CVPR 2018
Interactive systems
Video-based interfaces

Human joystick
NewsBreaker Live

Assistive technology systems
Camera Mouse
Boston College

YouTube Link
Vision for medical & neuroimages

Image guided surgery
MIT AI Vision Group

fMRI data
Golland et al.

Kristen Grauman
Safety & security

Navigation, driver safety

Monitoring pool (Poseidon)

Pedestrian detection
MERL, Viola et al.

Surveillance
Computer vision topics still being researched (from recent conferences and journals)
Figure 1. Summary of our approach. While standard image models jointly train an image feature extractor and a linear classifier to predict some label, CLIP jointly trains an image encoder and a text encoder to predict the correct pairings of a batch of (image, text) training examples. At test time the learned text encoder synthesizes a zero-shot linear classifier by embedding the names or descriptions of the target dataset’s classes.
Open-vocabulary object detection

Figure 2: An overview of using ViLD for open-vocabulary object detection. ViLD distills the knowledge from a pretrained open-vocabulary image classification model. First, the category text embeddings and the image embeddings of cropped object proposals are computed, using the text and image encoders in the pretrained classification model. Then, ViLD employs the text embeddings as the region classifier (ViLD-text) and minimizes the distance between the region embedding and the image embedding for each proposal (ViLD-image). During inference, text embeddings of novel categories are used to enable open-vocabulary detection.

How to learn from weak supervision

The elephant are about to march through them. The spiders themselves have a span as wide as a

But the love serenade is over once a dog arrives.

Australian camels appear sick and emaciated.

Tigers are one of the few cats that actually enjoy swimming.

Male koalas play no role in parenting.

About 50 animals have died in just three months, including this adult orangutan on the day we

Unlike mechanics, langurs are the friends of spotted deer.

There's a turf war going on and the koalas are losing. (dog)

The mayor has declined offers of assistance and expert advice from animal welfare groups. (elephant)
How to recognize objects in new modalities

Figure 6: Qualitative Results. YOLO running on sample artwork and natural images from the internet. It is mostly accurate although it does think one person is an airplane.
How to use models across countries

How to query vision-language models

![Image](image.png)

**Fig. 1** Prompt engineering vs Context Optimization (CoOp). The former needs to use a held-out validation set for words tuning, which is inefficient; the latter automates the process and requires only a few labeled images for learning.
How to query vision-language models

Figure 1: Schematic of the method. (Left) The standard method of a zero-shot open vocabulary image classification model (e.g., CLIP [Radford et al., 2021]). (Right) Our method of CuPL. First, an LLM generates descriptive captions for given class categories. Next, an open vocabulary model uses these captions as prompts for performing classification.

How to integrate modalities (audio)

Prior work: strong supervision with object detectors

Category label: "Guitar"
Audio waveform
Video with detected objects

Audio separation
Separated audio for the detected object

Ours: self-supervised without object detectors

Natural language query: "person playing a guitar"
Audio waveform
Video

Audio separation and localization
Separated audio and localized regions

Figure 1. We propose to separate and localize audio sources based on a natural language query, by learning to align the modalities on completely unlabeled videos. In comparison, prior audio-visual sound separation approaches require object label supervision.
How to represent everyday activities

Figure 1. Ego4D is a massive-scale egocentric video dataset of daily life activity spanning 74 locations worldwide. Here we see a snapshot of the dataset (5% of the clips, randomly sampled) highlighting its diversity in geographic location, activities, and modalities. The data includes social videos where participants consented to remain unblurred. See https://ego4d-data.org/fig1.html for interactive figure.
How to understand activities and intents
How to grade how well an activity is performed

"Stretch Hands"  "Lower Feet"

Quality of Action: 86.5 / 100
How to imagine motion in static images
How to decode physics from video

3D reconstruction
How to perform high-level reasoning

Figure 1. VISPROG is a modular and interpretable neuro-symbolic system for compositional visual reasoning. Given a few examples of natural language instructions and the desired high-level programs, VISPROG generates a program for any new instruction using in-context learning in GPT-3 and then executes the program on the input image(s) to obtain the prediction. VISPROG also summarizes the intermediate outputs into an interpretable visual rationale (Fig. 4). We demonstrate VISPROG on tasks that require composing a diverse set of modules for image understanding and manipulation, knowledge retrieval, and arithmetic and logical operations.

How to understand stories in film

Figure 1: Our MovieQA dataset contains 14,944 questions about 408 movies. It contains multiple sources of information: plots, subtitles, video clips, scripts, and DVS transcriptions. In this figure we show example QAs from The Matrix and localize them in the timeline.
How to understand roles in film

Video Clip

Legend
- Character
- Attribute
- Relationship
- Interaction
- Summary Int.
- Topic
- Reason
- Timestamp

Scene: Field Road
Situation: Bullying
Description:
As Jenny and Forrest are on the road, three boys start throwing rocks at Forrest. Jenny urges him to run from them. While Forrest runs, his leg braces fall apart.
How to understand media persuasion

Fig. 1: Example advertisements from our dataset that require challenging visual recognition and reasoning. Despite the potential applications of understanding the messages of ads, this problem has not been tackled in computer vision.

Understanding advertisements is more challenging than simply recognizing physical content from images, as ads employ a variety of strategies to persuade viewers.

We collect an advertisement dataset containing 64,832 images and 3,477 videos, each annotated by 3-5 human workers from Amazon Mechanical Turk.

Here are some sample annotations in our dataset.

### Image

<table>
<thead>
<tr>
<th></th>
<th>Topic</th>
<th>Strategy</th>
<th>20,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentiment</td>
<td>102,340</td>
<td>Symbol</td>
<td>64,131</td>
</tr>
<tr>
<td>Q+A Pair</td>
<td>202,090</td>
<td>Slogan</td>
<td>11,130</td>
</tr>
</tbody>
</table>

### Video

<table>
<thead>
<tr>
<th></th>
<th>Topic</th>
<th>Fun/Exciting</th>
<th>15,380</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentiment</td>
<td>17,345</td>
<td>English?</td>
<td>17,374</td>
</tr>
<tr>
<td>Q+A Pair</td>
<td>17,345</td>
<td>Effective</td>
<td>16,721</td>
</tr>
</tbody>
</table>

What’s being advertised in this image?
- Cars, automobiles

What sentiments are provoked in the viewer?
- Amused, Creative, Impressed, Youthful, Conscious

What strategies are used to persuade viewer?
- Symbolism, Contrast, Straightforward, Transferred qualities

What should the viewer do, and why should they do this?
- I should buy Volkswagen because it can hold a big bear.
- I should buy VW SUV because it can fit anything and everything in it.
- I should buy this car because it can hold everything I need.

More information available at [http://cs.pitt.edu/~kovashka/ads](http://cs.pitt.edu/~kovashka/ads)

Hussein et al., CVPR 2017
How to generate arbitrary content

Figure 1. Make-A-Scene: Samples of generated images from text inputs (a), and a text and scene input (b). Our method is able to both generate the scene (a, bottom left) and image, or generate the image from text and a simple sketch input (b, center).

How to reason and act

Das et al., CVPR 2018
How to use language models for robotics tasks

Figure 1: LLMs have not interacted with their environment and observed the outcome of their responses, and thus are not grounded in the world. SayCan grounds LLMs via value functions of pretrained skills, allowing them to execute real-world, abstract, long-horizon commands on robots.

Computer vision is not solved

- Deep learning makes excellent use of massive data (labeled for the task of interest?)
  - But it’s hard to understand how it does so, makes it hard to fix when it doesn’t work well
  - It doesn’t work well when massive data is not available and your task is different than tasks for which data is available
  - We can recognize objects with 97% accuracy but reasoning about relationships and intent is harder
Obstacles?

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition."

Read more about the history: Szeliski Sec. 1.2
Why is vision difficult?

- Ill-posed problem: real world much more complex than what we can measure in images
  - 3D $\rightarrow$ 2D
  - Motion $\rightarrow$ static
- Impossible to literally “invert” image formation process with limited information
  - Need information outside of this particular image to generalize what image portrays (e.g. to resolve occlusion)

Adapted from Kristen Grauman
What the computer gets

Why is this problematic?

Adapted from Kristen Grauman and Lana Lazebnik
Challenges: many nuisance parameters

Illumination  Object pose  Clutter

Occlusions  Intra-class appearance  Viewpoint

Think again about the pixels...
Challenges: intra-class variation

slide credit: Fei-Fei, Fergus & Torralba

CMOA Pittsburgh
Challenges: importance of context

slide credit: Fei-Fei, Fergus & Torralba
Challenges: Complexity

- Thousands to millions of pixels in an image
- 3,000-30,000 human recognizable object categories
- 30+ degrees of freedom in the pose of articulated objects (humans)
- Billions of images indexed by Google Image Search
- 1.424 billion smart camera phones sold in 2015
- About half of the cerebral cortex in primates is devoted to processing visual information [Felleman and van Essen 1991]
Challenges: Limited supervision

Unlabeled, multiple objects

Classes labeled, some clutter

Cropped to object, parts and classes

Kristen Grauman
Evolution of datasets

• Challenging problem $\rightarrow$ active research area

PASCAL:
20 categories, 12k images

ImageNet:
22k categories, 14mil images

Microsoft COCO:
80 categories, 300k images
Some Visual Recognition Problems: Why are they challenging?
Recognition: What objects do you see?

- building
- street
- balcony
- truck
- carriage
- horse
- table
- person
Detection: Where are the cars?
Activity: What is this person doing?
Scene: Is this an indoor scene?
Instance: Which city? Which building?
Visual question answering: Why is there a carriage in the street?
What to do about variance?
(albino koalas)

• You can’t always do anything about it
• Basic assumption: training and test set are sampled from the same distribution
• Practically speaking: Make sure training set is representative
• Tricks (e.g. data preprocessing) make learning stable
• You can try to artificially expand variance in training set
• You can detect outliers, report confidence of prediction
• Still an open problem
Problem with categorization
(Borges' Animals)

“These ambiguities, redundancies and deficiencies recall those that Dr. Franz Kuhn attributes to a certain Chinese dictionary entitled *The Celestial Emporium of Benevolent Knowledge*. In its remote pages it is written that animals can be divided into (a) those belonging to the Emperor, (b) those that are embalmed, (c) those that are tame, (d) pigs, (e) sirens, (f) imaginary animals, (g) wild dogs, (h) those included in this classification, (i) those that are crazy-acting, (j) those that are uncountable, (k) those painted with the finest brush made of camel hair, (l) miscellaneous, (m) those which have just broken a vase, and (n) those which, from a distance, look like flies.”

Overview of topics
Features and filters

- Describing and transforming textures, colors, edges

Kristen Grauman
Features and filters

- Detecting distinctive and repeatable features
- Describing images with local statistics
Grouping and fitting

- Clustering, segmentation, fitting; what parts belong together?

Kristen Grauman

[fig from Shi et al]
Multiple views

- Multi-view geometry, matching, stereo vision

Hartley and Zisserman

Lowe

Fei-Fei Li

Kristen Grauman
Image categorization

- Fine-grained recognition

Generalist  Insect catching  Grain eating  Coniferous-seed eating  Nectar feeding
Chiseling  Dip netting  Surface skimming  Scything  Probing
Aerial fishing  Pursuit fishing  Scavenging  Raptorial  Filter feeding

Visipedia Project
Image categorization

• Material recognition

[Bell et al. CVPR 2015]
Image categorization

- Image style recognition

<table>
<thead>
<tr>
<th>HDR</th>
<th>Macro</th>
<th>Baroque</th>
<th>Roccoco</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vintage</td>
<td>Noir</td>
<td>Northern Renaissance</td>
<td>Cubism</td>
</tr>
<tr>
<td>Minimal</td>
<td>Hazy</td>
<td>Impressionism</td>
<td>Post-Impressionism</td>
</tr>
<tr>
<td>Long Exposure</td>
<td>Romantic</td>
<td>Abs. Expressionism</td>
<td>Color Field Painting</td>
</tr>
</tbody>
</table>

Flickr Style: 80K images covering 20 styles.

WikiPaintings: 85K images for 25 art genres.

[Karayev et al. BMVC 2014]
Visual recognition and SVMs

- Recognizing objects and categories, learning techniques

Adapted from Kristen Grauman
Convolutional neural networks (CNNs)

- State-of-the-art on many recognition tasks

Krizhevsky et al., NIPS 2012

Yosinski et al., ICML DL workshop 2015
Recurrent neural networks (RNNs)

- Sequence processing, e.g. question answering

Wu et al., CVPR 2016
Motion and tracking

- Tracking objects, video analysis
Pose and actions

- Automatically annotating human pose (joints)
- Recognizing actions in first-person video
Unsupervised learning
<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>A</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
</tbody>
</table>

Unsupervised learning

Doersch et al., ICCV 2015
Linear algebra review

See [http://cs229.stanford.edu/section/cs229-linalg.pdf](http://cs229.stanford.edu/section/cs229-linalg.pdf) for more
What are images? (in Matlab)

• Matlab treats images as matrices of numbers
• To proceed, let’s talk very briefly about how images are formed
Image formation
Digital images

FIGURE 2.17 (a) Continuous image projected onto a sensor array. (b) Result of image sampling and quantization.

- **Sample** the 2D space on a regular grid
- **Quantize** each sample (round to nearest integer)
Digital images

- **Sample** the 2D space on a regular grid
- **Quantize** each sample (round to nearest integer)
- What does quantizing signal look like?

Image thus represented as a matrix of integer values.

Adapted from S. Seitz
Digital color images

Bayer filter

© 2000 How Stuff Works
Digital color images

Color images, RGB color space:

Split image into three channels

Adapted from Kristen Grauman
Images in Matlab

- Color images represented as a matrix with multiple channels (=1 if grayscale)
- Suppose we have a NxM RGB image called “im”
  - \(im(1,1,1)\) = top-left pixel value in R-channel
  - \(im(y, x, b)\) = \(y\) pixels down, \(x\) pixels to right in the \(b\)th channel
  - \(im(N, M, 3)\) = bottom-right pixel in B-channel
- `imread(filename)` returns a uint8 image (values 0 to 255)
  - Convert to double format with `double` or `im2double`

<table>
<thead>
<tr>
<th></th>
<th>row</th>
<th>column</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.92</td>
<td>0.93</td>
<td>0.94</td>
</tr>
<tr>
<td>0.95</td>
<td>0.89</td>
<td>0.82</td>
</tr>
<tr>
<td>0.89</td>
<td>0.72</td>
<td>0.51</td>
</tr>
<tr>
<td>0.96</td>
<td>0.95</td>
<td>0.88</td>
</tr>
<tr>
<td>0.71</td>
<td>0.81</td>
<td>0.81</td>
</tr>
<tr>
<td>0.49</td>
<td>0.62</td>
<td>0.60</td>
</tr>
<tr>
<td>0.86</td>
<td>0.84</td>
<td>0.74</td>
</tr>
<tr>
<td>0.96</td>
<td>0.67</td>
<td>0.54</td>
</tr>
<tr>
<td>0.69</td>
<td>0.49</td>
<td>0.56</td>
</tr>
<tr>
<td>0.79</td>
<td>0.73</td>
<td>0.90</td>
</tr>
<tr>
<td>0.91</td>
<td>0.94</td>
<td>0.89</td>
</tr>
</tbody>
</table>

Adapted from Derek Hoiem
Vectors and Matrices

• Vectors and matrices are just collections of ordered numbers that represent something: movements in space, scaling factors, word counts, movie ratings, pixel brightnesses, etc.

• We’ll define some common uses and standard operations on them.
Vector

• A column vector \( \mathbf{v} \in \mathbb{R}^{n\times 1} \) where
  \[
  \mathbf{v} = \begin{bmatrix}
  v_1 \\
  v_2 \\
  \vdots \\
  v_n
  \end{bmatrix}
  \]

• A row vector \( \mathbf{v}^T \in \mathbb{R}^{1\times n} \) where
  \[
  \mathbf{v}^T = \begin{bmatrix}
  v_1 & v_2 & \ldots & v_n
  \end{bmatrix}
  \]

\( T \) denotes the transpose operation
Vector

• You’ll want to keep track of the orientation of your vectors when programming in MATLAB.
• You can transpose a vector $\mathbf{v}$ in MATLAB by writing $\mathbf{v}'$. 
Vectors have two main uses

- Vectors can represent an offset in 2D or 3D space
- Points are just vectors from the origin
- Data can also be treated as a vector
- Such vectors don’t have a geometric interpretation, but calculations like “distance” still have value
Norms

• L1 norm

\[ \| x \|_1 := \sum_{i=1}^{n} |x_i| \]

• L2 norm

\[ \| x \| := \sqrt{x_1^2 + \cdots + x_n^2} \]

• \( L^p \) norm (for real numbers \( p \geq 1 \))

\[ \| x \|_p := \left( \sum_{i=1}^{n} |x_i|^p \right)^{1/p} \]
Distances

• L1 (Manhattan) distance

\[ d_1(p, q) = \|p - q\|_1 = \sum_{i=1}^{n} |p_i - q_i|, \]

• L2 (Euclidean) distance

\[ d(p, q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \]

\[ d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_i - q_i)^2 + \cdots + (p_n - q_n)^2}. \]
Example: Feature representation

• A vector representing measurable characteristics of a data sample we have

• E.g. a glass of juice can be represented via its color = \{yellow=1, red=2, green=3, purple=4\} and taste = \{sweet=1, sour=2\}

• A given glass \( i \) can be represented as a vector: \( \mathbf{x}_i = [3 2] \) represents *green, sour* juice

• For \( D \) features, this defines a \( D \)-dimensional space where we can measure similarity between samples
Example: Feature representation

E.g. a glass of juice can be represented via its color = \{yellow=1, red=2, green=3, purple=4\} and taste = \{sweet=1, sour=2\}

L2 distance:
\[
d(x_1, x_2) = \sqrt{4+0} = 2 \\
d(x_1, x_3) = \sqrt{0+1} = 1 \\
d(x_2, x_3) = \sqrt{4+1} = 2.2
\]

L1 distance:
\[
d(x_1, x_2) = 2+0 = 2 \\
d(x_1, x_3) = 0+1 = 1 \\
d(x_2, x_3) = 2+1 = 3
\]
Matrix

• A matrix $A \in \mathbb{R}^{m \times n}$ is an array of numbers with size $m \downarrow$ by $n \rightarrow$, i.e. $m$ rows and $n$ columns.

$$A = \begin{bmatrix}
    a_{11} & a_{12} & a_{13} & \ldots & a_{1n} \\
    a_{21} & a_{22} & a_{23} & \ldots & a_{2n} \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    a_{m1} & a_{m2} & a_{m3} & \ldots & a_{mn}
\end{bmatrix}$$

• If $m = n$, we say that $A$ is square.
Matrix Operations

• Addition

\[
\begin{bmatrix}
  a & b \\
  c & d \\
\end{bmatrix}
+ \begin{bmatrix}
  1 & 2 \\
  3 & 4 \\
\end{bmatrix}
= \begin{bmatrix}
  a + 1 & b + 2 \\
  c + 3 & d + 4 \\
\end{bmatrix}
\]

– Can only add a matrix with matching dimensions, or a scalar.

\[
\begin{bmatrix}
  a & b \\
  c & d \\
\end{bmatrix}
+ 7
= \begin{bmatrix}
  a + 7 & b + 7 \\
  c + 7 & d + 7 \\
\end{bmatrix}
\]

• Scaling

\[
\begin{bmatrix}
  a & b \\
  c & d \\
\end{bmatrix}
\times 3
= \begin{bmatrix}
  3a & 3b \\
  3c & 3d \\
\end{bmatrix}
\]
Matrix Multiplication

- Let X be an $axb$ matrix, Y be an $bxc$ matrix
- Then $Z = X*Y$ is an $axc$ matrix
- Second dimension of first matrix, and first dimension of second matrix have to be the same, for matrix multiplication to be possible
- Practice: Let X be an 10x5 matrix. Let’s factorize it into 3 matrices...
Matrix Multiplication

• The product AB is:

• Each entry in the result is (that row of A) dot product with (that column of B)
Matrix Multiplication

• Example:

\[
\begin{bmatrix}
0 & 2 \\
4 & 6
\end{bmatrix}
\begin{bmatrix}
1 & 3 \\
5 & 7
\end{bmatrix}
\]

\[
\begin{bmatrix}
0 \cdot 3 + 2 \cdot 7 = 14
\end{bmatrix}
\]

– Each entry of the matrix product is made by taking the dot product of the corresponding row in the left matrix, with the corresponding column in the right one.
Inner (Dot) Product

- Multiply corresponding entries of two vectors and add up the result
  \[ x^T y = \begin{bmatrix} x_1 & \ldots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \sum_{i=1}^{n} x_i y_i \quad \text{scalar} \]
- \( x \cdot y \) is also \( |x| \cdot |y| \cdot \cos(\text{angle between } x \text{ and } y) \)
- If \( B \) is a unit vector, then \( A \cdot B \) gives the length of \( A \) which lies in the direction of \( B \) (projection)

(if \( B \) is unit-length hence norm is 1)
Different Types of Product

• \( x, y \) = column vectors (nx1)
• \( X, Y \) = matrices (mxn)
• \( x, y \) = scalars (1x1)

• \( x^T y = x \cdot y \) = inner product (1xn x nx1 = scalar)
• \( x \otimes y = xy^T \) = outer product (nx1 x 1xn = matrix)

• \( X * Y \) = matrix product
• \( X .* Y \) = element-wise product
Matrix Operations

• Transpose – flip matrix, so row 1 becomes column 1

\[
\begin{bmatrix}
0 & 1 & \ldots
\end{bmatrix}
\]

\[
\begin{bmatrix}
0 & 1 \\
2 & 3 \\
4 & 5 \\
\end{bmatrix}^T =
\begin{bmatrix}
0 & 2 & 4 \\
1 & 3 & 5 \\
\end{bmatrix}
\]

• A useful identity:

\[
(ABC)^T = C^T B^T A^T
\]
Matrix Operations

• MATLAB example:

\[ AX = B \]

\[ A = \begin{bmatrix} 2 & 2 \\ 3 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \]

\[
>> x = A\backslash B \\
x = \\
\begin{bmatrix}
1.0000 \\
-0.5000
\end{bmatrix}
\]
Matrix Operation Properties

• Matrix addition is commutative and associative
  – $A + B = B + A$
  – $A + (B + C) = (A + B) + C$

• Matrix multiplication is associative and distributive but *not* commutative
  – $A(B*C) = (A*B)C$
  – $A(B + C) = A*B + A*C$
  – $A*B \neq B*A$
Special Matrices

• Identity matrix $I$
  – Square matrix, 1’s along diagonal, 0’s elsewhere
  – $I \cdot [\text{another matrix}] = [\text{that matrix}]

• Diagonal matrix
  – Square matrix with numbers along diagonal, 0’s elsewhere
  – A diagonal $\cdot [\text{another matrix}]$ scales the rows of that matrix
Special Matrices

• Symmetric matrix

\[ A^T = A \]

\[
\begin{bmatrix}
1 & 2 & 5 \\
2 & 1 & 7 \\
5 & 7 & 1
\end{bmatrix}
\]
Matlab
Matlab tutorial

http://www.cs.pitt.edu/~kovashka/cs1674_fa18/tutorial.m
http://www.cs.pitt.edu/~kovashka/cs1674_fa18/myfunction.m
http://www.cs.pitt.edu/~kovashka/cs1674_fa18/myotherfunction.m

Please cover whatever we don’t finish at home.
Other tutorials and exercises

- [https://people.cs.pitt.edu/~milos/courses/cs2750/Tutorial/](https://people.cs.pitt.edu/~milos/courses/cs2750/Tutorial/)
- [http://www.math.udel.edu/~braun/M349/Matlab_probs2.pdf](http://www.math.udel.edu/~braun/M349/Matlab_probs2.pdf)
- [http://www.facstaff.bucknell.edu/maneval/help211/basicexercises.html](http://www.facstaff.bucknell.edu/maneval/help211/basicexercises.html)
  - Do Problems 1-8, 12
  - Most also have solutions
  - Ask the TA if you have any problems