Introduction to Computer Vision and Image Classification

Danna Gurari
University of Colorado Boulder
Spring 2024

https://dannagurari.colorado.edu/course/neural-networks-and-deep-learning-spring-2024/
Review

• Last lecture:
  • Neural Networks for Spatial Data
  • History of Convolutional Neural Networks (CNNs)
  • CNNs – Convolutional Layers
  • CNNs – Pooling Layers
  • Programming tutorial

• Assignments (Canvas)
  • Problem set 2 due earlier today
  • Lab assignment 2 due in 1.5 weeks

• Questions?
Today’s Topics

• Computer vision

• Era of dataset challenges

• MNIST challenge winner: LeNet

• Pioneering ImageNet challenge winner: AlexNet (deeper learning)

• Programming tutorial
Today’s Topics

• Computer vision

• Era of dataset challenges

• MNIST challenge winner: LeNet

• Pioneering ImageNet challenge winner: AlexNet (deeper learning)

• Programming tutorial
Computer Vision: Computers that “See”

- Self-driving cars
- Exploration on Mars
- Guided surgery
- Visual assistance for people who are blind
- Security
Why Discuss Computer Vision With CNNs?

- CNNs have a strong track record for vision problems
- Visual data’s representation (i.e., spatial data) is naturally suited for CNNs
Historical Context: Origins of Computer Vision

1847 Gradient descent
1945 First programmable machine
1950 Turing test
1956-1957 Perceptron
1959 Machine learning
1959 Wave 3: rise of "deep learning"
2012

First digital image:
176 x 176 pixels
Historical Context: Origins of Computer Vision

- 1847: Gradient descent
- 1945: First programmable machine
- 1950: Turing test
- 1956: Artificial Intelligence (AI)
- 1957: Turing test
- 1959: First digital image; Perceptron
- 1966: Birth of computer vision

Excerpt from a document:

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. 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".
Historical Context: Origins of Computer Vision

1847 1945 1950 1956 1957 1959 1966

- Gradient descent
- First programmable machine
- Turing test
- AI
- First digital image; Perceptron
- Machine learning
- Birth of computer vision

Turing test: design “computer vision” that is indistinguishable from “human vision”
Computer Vision in Context

Artificial Intelligence (machines that do “intelligent” things)

Deep learning (machines that learn to do “intelligent” things using neural networks)

Computer Vision (machines that “see”)

Course scope
Key Challenge: Replicate Human Vision for So Much Variation for So Many Tasks!

- Object recognition
- Object detection
- Segmentation
- Image captioning
- Visual question answering
- Object tracking
- Subjective problems
- And more...
Key Challenge: Replicate Human Vision for So Much Variation for So Many Tasks!

- Object recognition
- Object detection
- Segmentation
- Image captioning
- Visual question answering
- Object tracking
- Subjective problems
- And more...

e.g., take a picture of an object and find where to buy it
Key Challenge: Replicate Human Vision for So Much Variation for So Many Tasks!

- Object recognition
- Object detection
- Segmentation
- Image captioning
- Visual question answering
- Object tracking
- Subjective problems
- And more...

e.g., detect faces to tag
Key Challenge: Replicate Human Vision for So Much Variation for So Many Tasks!

- Object recognition
- Object detection
- Segmentation
- Image captioning
- Visual question answering
- Object tracking
- Subjective problems
- And more...

e.g., rotoscoping

https://www.starnow.co.uk/ahmedmohammed1/photos/4650871/before-and-after-rotoscopinggreen-screening
Key Challenge: Replicate Human Vision for So Much Variation for So Many Tasks!

- Object recognition
- Object detection
- Segmentation
- Image captioning
- Visual question answering
- Object tracking
- Subjective problems
- And more...

e.g., Microsoft Power Point
Key Challenge: Replicate Human Vision for So Much Variation for So Many Tasks!

- Object recognition
- Object detection
- Segmentation
- Image captioning
- Visual question answering
- Object tracking
- Subjective problems
- And more...

e.g., BeSpecular
https://www.lionessesofafrica.com/blog/2015/2/15/the-startup-story-of-stephanie-cowper
Key Challenge: Replicate Human Vision for So Much Variation for So Many Tasks!

- Object recognition
- Object detection
- Segmentation
- Image captioning
- Visual question answering
- Object tracking
- Subjective problems
- And more...

e.g., track bowling ball path

e.g., calculate bat speed
Key Challenge: Replicate Human Vision for So Much Variation for So Many Tasks!

- Object recognition
- Object detection
- Segmentation
- Image captioning
- Visual question answering
- Object tracking
- Subjective problems
- And more…
Key Challenge: Replicate Human Vision for So Much Variation for **So Many Tasks**!

- Object recognition
- Object detection
- Segmentation
- Image captioning
- Visual question answering
- Object tracking
- Subjective problems
- And more...
Key Challenge: Replicate Human Vision for So Much Variation for So Many Tasks!

- Illumination
- Object pose
- Clutter
- Occlusions
- Intra-class appearance
- Viewpoint
Today’s Topics

• Computer vision

• Era of dataset challenges

• MNIST challenge winner: LeNet

• Pioneering ImageNet challenge winner: AlexNet (deeper learning)

• Programming tutorial
Datasets tended to be relatively small (e.g., 10s or 100s of examples)
Status Quo Until 2012: Datasets

- Authors created datasets primarily with their cameras, purchasing from companies, or downloading images from the Internet

- What’s wrong with this approach?
  - Unable to perform “fair” comparison between algorithms
  - Lacks a community around a shared goal
Datasets tend to be large (e.g., thousands to billions of examples)
Status Quo Since 2012

Datasets tend to be large (e.g., thousands to billions of examples)

What do you think prompted this shift to large-scale datasets?
Research Since 2012: Dataset Challenges

Create AI Challenges (Tests) Paired With Public Leaderboards to Track Progress
Research Since 2012: Dataset Challenges

Key components:

1. Publicly-shared test examples without ground truth answers for evaluation

2. Metrics for evaluating algorithm predictions, implemented in an evaluation server

3. Publicly-shared examples with “ground truth” answers to support training and validation
Research Since 2012: Dataset Challenges

Many public dataset challenges and datasets:

- Google Dataset Search
- Kaggle
- Amazon’s AWS datasets
- UC Irvine Machine Learning Repository
- Quora.com
- Reddit
- Dataportals.org
- Opendatamonitor.eu
- Quandl.com
Today’s Topics

• Computer vision

• Era of dataset challenges

• MNIST challenge winner: LeNet

• Pioneering ImageNet challenge winner: AlexNet (deeper learning)

• Programming tutorial
Historical Context: Inspiration

- 1847: Gradient descent
- 1945: First programmable machine
- 1950: Turing test
- 1956: AI
- 1957: Perceptron
- 1959: Machine learning
- 1960: First effective learning strategy
- 1980: Neocognitron
- 1986: Backpropagation for CNNs
- 1989: MNIST, LeNet
- 1998: Wave 3: rise of “deep learning”
Key contribution: showing how to perform backpropagation for CNNs to enable learning thereby eliminating the need for hand-crafted filters
MNIST Dataset Challenge

• **Goal**: classify digit as 0, 1, ..., or 9
• **Evaluation metric**: accuracy (% correct)
• **Dataset**: 60,000 training and 10,000 test examples, pre-processed to be centered and same dimension; writers were different in the two sets
• **Source**: images collected by NIST from a total of 500 Census Bureau employees and high school students

Figure source: [https://commons.wikimedia.org/w/index.php?curid=64810040](https://commons.wikimedia.org/w/index.php?curid=64810040)
LeNet: Architecture (like Neocognitron, has alternating convolutional layers and pooling layers)

Multi-layer neural network

Y. LeCun; L. Bottou; Y. Bengio; P. Haffner; Gradient-based learning applied to document recognition; 1998

tanh is used as the activation function
LeNet: Architecture (like Neocognitron, has alternating **convolutional** layers and **pooling** layers)

How many possible output values does this network predict?

Multi-layer neural network

Y. Lecun; L. Bottou; Y. Bengio; P. Haffner; Gradient-based learning applied to document recognition; 1998
LeNet: Architecture (like Neocognitron, has alternating **convolutional** layers and **pooling** layers)

How many filters are between the input and hidden layer 1?

Y. Lecun; L. Bottou; Y. Bengio; P. Haffner; Gradient-based learning applied to document recognition; 1998
LeNet: Architecture (like Neocognitron, has alternating convolutional layers and pooling layers)

What size of a neighborhood is used for this pooling layer?

Y. Lecun; L. Bottou; Y. Bengio; P. Haffner; Gradient-based learning applied to document recognition; 1998
Training Procedure Approach (Key Novelty)

- Repeat until stopping criterion met:
  1. **Forward pass**: propagate training data through model to make prediction
  2. Quantify the dissatisfaction with a model’s results on the training data
  3. **Backward pass**: using predicted output, calculate gradients backward to assign blame to each model parameter
  4. Account for weight sharing by using average of all connections for a parameter
  5. Update each parameter using calculated gradients

Figure from: Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, Jeffrey Mark Siskind; Automatic Differentiation in Machine Learning: a Survey; 2018
Training Procedure Approach (Key Novelty)

• Repeat until stopping criterion met:

1. **Forward pass**: propagate training data through model to make prediction
2. Quantify the dissatisfaction with a model’s results on the training data
3. **Backward pass**: using predicted output, calculate gradients backward to assign blame to each model parameter
4. Account for weight sharing by using average of all connections for a parameter
5. Update each parameter using calculated gradients

Still obtain an error surface, $E$, based on the chosen objective function (e.g., using mean squared error, cross entropy loss)

Training Procedure Approach (Key Novelty)

• Repeat until stopping criterion met:
  1. **Forward pass**: propagate training data through model to make prediction
  2. Quantify the dissatisfaction with a model’s results on the training data
  3. **Backward pass**: using predicted output, calculate gradients backward to assign blame to each model parameter
  4. Account for weight sharing by using average of all connections for a parameter
  5. Update each parameter using calculated gradients

A gradient is computed for each value in each convolutional filter (i.e., model weight) as well as all bias terms


Training Procedure Approach (Key Novelty)

• Repeat until stopping criterion met:

1. **Forward pass**: propagate training data through model to make prediction

2. Quantify the dissatisfaction with a model’s results on the training data

3. **Backward pass**: using predicted output, calculate gradients backward to assign blame to each model parameter

4. Account for weight sharing by using average of all connections for a parameter

5. Update each parameter using calculated gradients

LeNet vs Neocognitron

**Neocognitron: Hand-crafted filters**
- Feature Extraction
- Prediction

**LeNet: Learned filters**
- Feature Extraction
- Prediction

INPUT

OUTPUT

9

9
LeNet Analysis

How many epochs are needed for training to converge?

Why might overfitting not arise with more training?
- Learning rate too large for the model to settle in a local minimum (instead oscillated randomly)

Y. Lecun; L. Bottou; Y. Bengio; P. Haffner; Gradient-based learning applied to document recognition; 1998
LeNet Analysis

All 82 mislabeled examples (correct answer on left, predicted answer on right):

Why might the model be making mistakes?
- Insufficient representation in the training data
- Ambiguity

Y. Lecun; L. Bottou; Y. Bengio; P. Haffner; Gradient-based learning applied to document recognition; 1998
LeNet, designed on the MNIST Challenge, was used to read over 10% of checks in North America in the 1990s, reading millions of checks every month.
Today’s Topics

• Computer vision

• Era of dataset challenges

• MNIST challenge winner: LeNet

• Pioneering ImageNet challenge winner: AlexNet (deeper learning)

• Programming tutorial
Historical Context

- 1847: Gradient descent
- 1945: First programmable machine
- 1950: Turing test
- 1956: AI
- 1957: Perceptron
- 1959: Machine learning
- 1960: Neocognitron
- 1969: Neural networks with effective learning strategy
- 1980: Backpropagation for CNNs
- 1985: MNIST, LeNet
- 1986: ImageNet: catalyst for 3rd wave
- 1989: Wave 3: rise of “deep learning”
ImageNet: Predict Category from 1000 Options

Is this a multi-label or a multi-class classification problem?

- **Evaluation metric**: % correct (top-1 and top-5 predictions)
- **Dataset**: ~1.5 million images
- **Source**: images scraped from search engines, such as Flickr, and labeled by crowdworkers

ImageNet vs MNIST

• 3D objects in natural backgrounds
• Many more categories
Rise of “Deep Learning”: Deeper NNs

Progress of models on ImageNet (Top 5 Error)

Scaling up the dataset led to unexpected, unprecedented improvements from the “deep learning” model, AlexNet

Figure Source: https://www.edge-ai-vision.com/2018/07/deep-learning-in-five-and-a-half-minutes/
(Model Named After First Author)

(2012, Neurips)

ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky  
University of Toronto  
kriz@cs.utoronto.ca

Ilya Sutskever  
University of Toronto  
ilya@cs.utoronto.ca

Geoffrey E. Hinton  
University of Toronto  
hinton@cs.utoronto.ca
Key Issue Addressed by AlexNet: Impractically, Slow Training Because of Vanishing Gradients

- Repeat until stopping criterion met:
  1. **Forward pass**: propagate training data through model to make prediction
  2. Quantify the dissatisfaction with a model's results on the training data
  3. **Backward pass**: using predicted output, calculate gradients backward to assign blame to each model parameter
  4. Update each parameter using calculated gradients

\[
W_x = W_x - \alpha \left( \frac{\partial \text{Error}}{\partial W_x} \right)
\]

Figure from: Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, Jeffrey Mark Siskind; Automatic Differentiation in Machine Learning: a Survey; 2018
Key Issue Addressed by AlexNet: Impractically, Slow Training Because of Vanishing Gradients

Recall activation functions and their derivatives:

Ranges from 0 to 0.25

Vanishing Gradient Problem (e.g., sigmoid)

• Toy example:

• Error Derivative with respect to weight w1:

\[
\frac{\partial \text{error}}{\partial w_1} = \frac{\partial \text{error}}{\partial \text{output}} \cdot \frac{\partial \text{output}}{\partial \text{hidden}_2} \cdot \frac{\partial \text{hidden}_2}{\partial \text{hidden}_1} \cdot \frac{\partial \text{hidden}_1}{\partial w_1}
\]

Problem: What happens as you multiply more numbers smaller than 1?
Gradient decreases as further from the last layer... and so weights barely change at training!

https://ayearofai.com/rohan-4-the-vanishing-gradient-problem-ec68f76ffb9b
Vanishing Gradient Problem (e.g., sigmoid)

Smallest gradients at earliest layers make them slowest to train, yet later layers depend on those earlier layers to do something useful; consequently, NNs struggle with garbage in means garbage out.
Idea: Use ReLU Activation Function

Use activation functions with derivative value equal to 1 (i.e., 1x1x1... doesn’t vanish)

AlexNet Architecture: Similar to LeNet But With More Convolutional and Pooling Layers

Source: https://www.learnopencv.com/wp-content/uploads/2018/05/AlexNet-1.png
AlexNet Architecture

Input: RGB image resized to fixed input size
Output: 1000 class probabilities (sums to 1)
Convolutional layers: 5 layers
Pooling layers: 3 layers
Fully-connected layers: 3 layers

Implementation detail: subtract mean image from training data to center input.

Source: https://www.learnopencv.com/wp-content/uploads/2018/05/AlexNet-1.png
How many layers have model parameters that need to be learned?
AlexNet Architecture

Altogether, 60 million model parameters must be learned!

Source: https://www.learnopencv.com/wp-content/uploads/2018/05/AlexNet-1.png
AlexNet Architecture

Most parameters come from the fully connected layers

Few parameters come from the convolutional layers

Altogether, 60 million model parameters must be learned!
Risk: Overfitting Due to 60 Million Parameters

• Remedies to address AlexNet’s large representational capacity
  1. Data augmentation: add more training data; e.g., intuitively,
Risk: Overfitting Due to 60 Million Parameters

• Remedies to address AlexNet’s large representational capacity
  1. Data augmentation
     1. Random patches and their mirror images (2048x more data)
     2. Adjust RGB channels (using PCA to add multiples of principal components)

Figure Source: https://learnopencv.com/understanding-alexnet/
Risk: Overfitting Due to 60 Million Parameters

• Remedies to address AlexNet’s large representational capacity
  1. Data augmentation
     1. Random patches and their mirror images (2048x more data)
     2. Adjust RGB channels (using PCA to add multiples of principal components)
  2. Dropout: 50% of nodes for first two fully connected layers (note, dropout was introduced in the same year also from the authors of AlexNet!)

AlexNet Training: 90 Epochs on 2 GPUs

Repeat until stopping criterion met:

1. **Forward pass**: propagate training data through model to make prediction

2. Quantify the dissatisfaction with a model’s results on the training data

3. **Backward pass**: using predicted output, calculate gradients backward to assign blame to each model parameter

4. Update each parameter using calculated gradients

Uses cross entropy loss
AlexNet Analysis

8 examples of predictions, correct and incorrect

When/why might the model succeed?
- Single well-defined object (even if off-centered)

When/why might the model fail?
- Ambiguity
- Similar categories

AlexNet: Inspecting What It Learned
AlexNet: Inspecting What It Learned (96 Filters)

Model learned filters that select based on frequency, orientation, and color! (aligns with Hubel & Weisel’s findings for how vision systems work)
Today’s Topics

• Computer vision

• Era of dataset challenges

• MNIST challenge winner: LeNet

• Pioneering ImageNet challenge winner: AlexNet (deeper learning)

• Programming tutorial
Today’s Topics

• Computer vision

• Era of dataset challenges

• MNIST challenge winner: LeNet

• Pioneering ImageNet challenge winner: AlexNet (deeper learning)

• Programming tutorial
The End