Visual AI

Deep learning models for Computer Graphics and Computer Vision

CPSC 532R/533R – 2019/2020 Term 2

Helge Rhodin

Visual AI

CPSC 532R/533R – 2019/2020 Term 2

Lecture 1. Overview and programming environment

Helge Rhodin

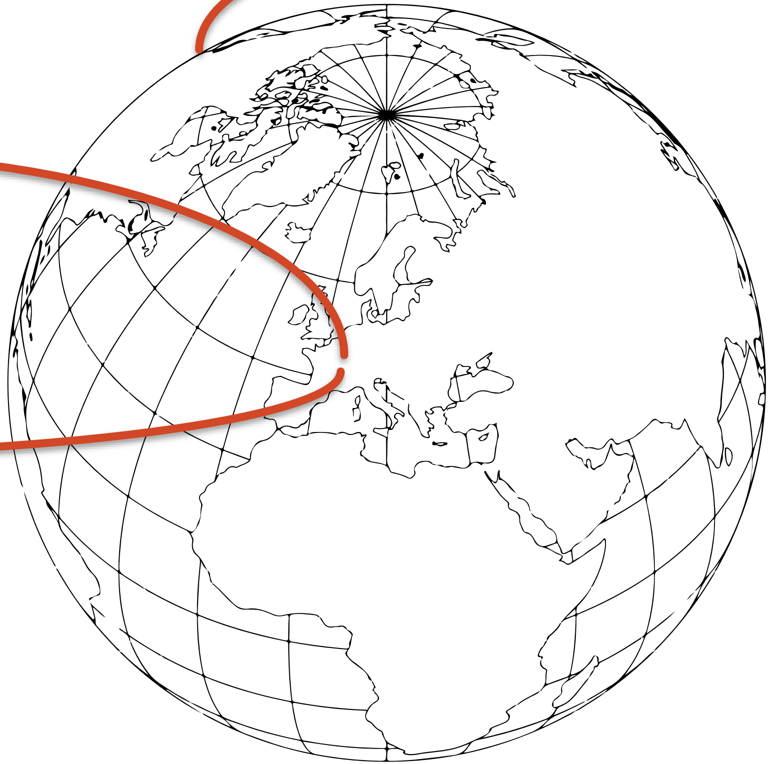
Helge Rhodin

Prof. at UBC

MSc and BSc
Saarland University, Germany

PhD
Max-Plank institute for Informatics

Lecturer and postdoc
EPFL, Switzerland



Organization

Instructor:

Helge Rhodin

rhodin@cs.ubc.ca

Office hours:

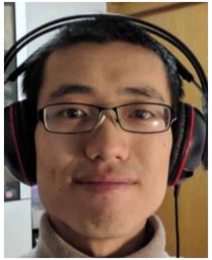
Wednesday 11 am – noon

Room ICCS X653

Teaching assistant:

Yuchi Zhang

yuchi45@cs.ubc.ca



Office hours:

Tuesday 1 pm – 2 pm room

Room ICCS X341

Course Website

Curriculum https://www.cs.ubc.ca/~rhodin/20_CPSC_532R_533R/

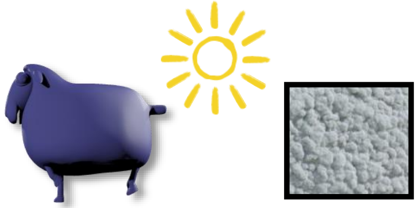
Forum https://piazza.com/ubc.ca/winterterm22019/cs532533

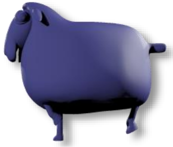
Computer graphics, computer vision and machine learning

Computer graphics

Computer vision

Scene model
(mesh, light source, texture, …)





3D mesh

$f : \theta \mapsto \mathbf{I}$

$f^{-1} : \mathbf{I} \mapsto \theta$

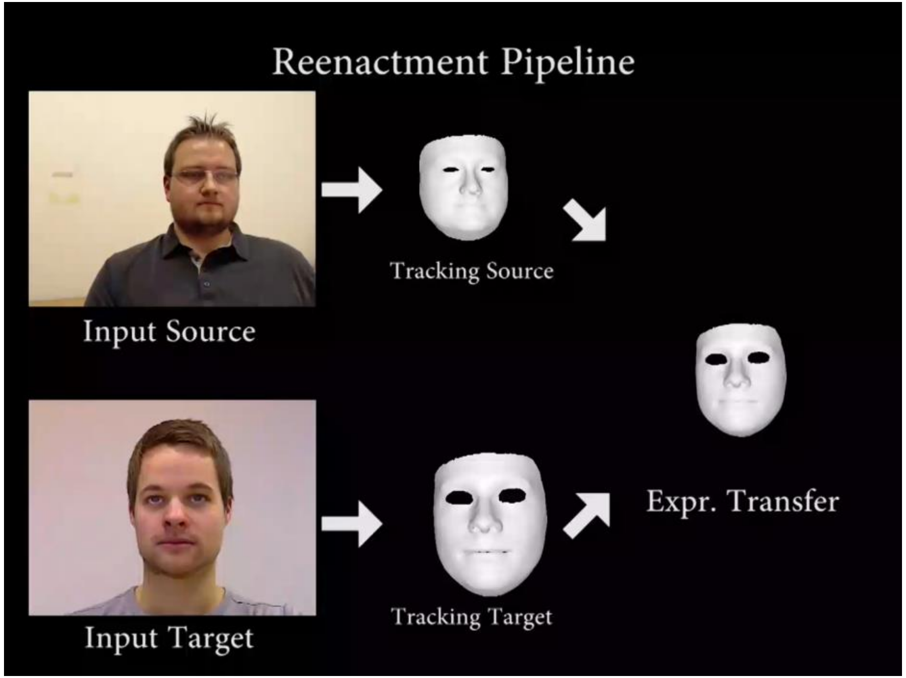
Image
(pixel intensities)

Image

Human pose estimation

[VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera, SIGGRAPH 2017]

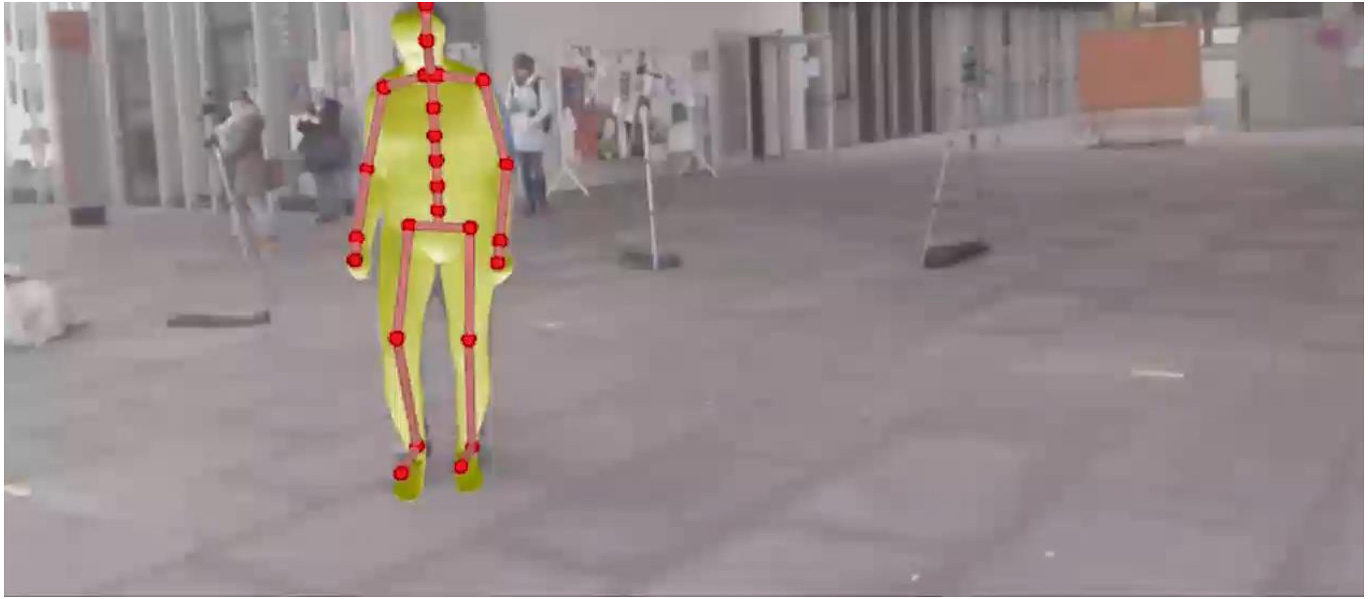
Facial reenactment



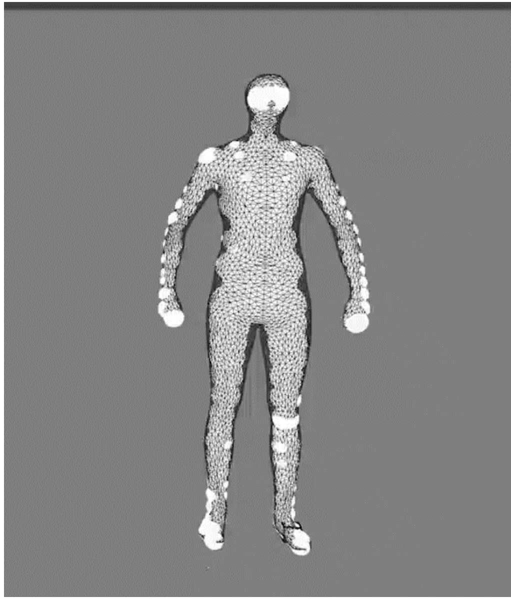
[Face2Face: Real-time Face Capture and Reenactment of RGB Videos, CVPR 2016]

Geometry based, no machine learning!

Generative human body models



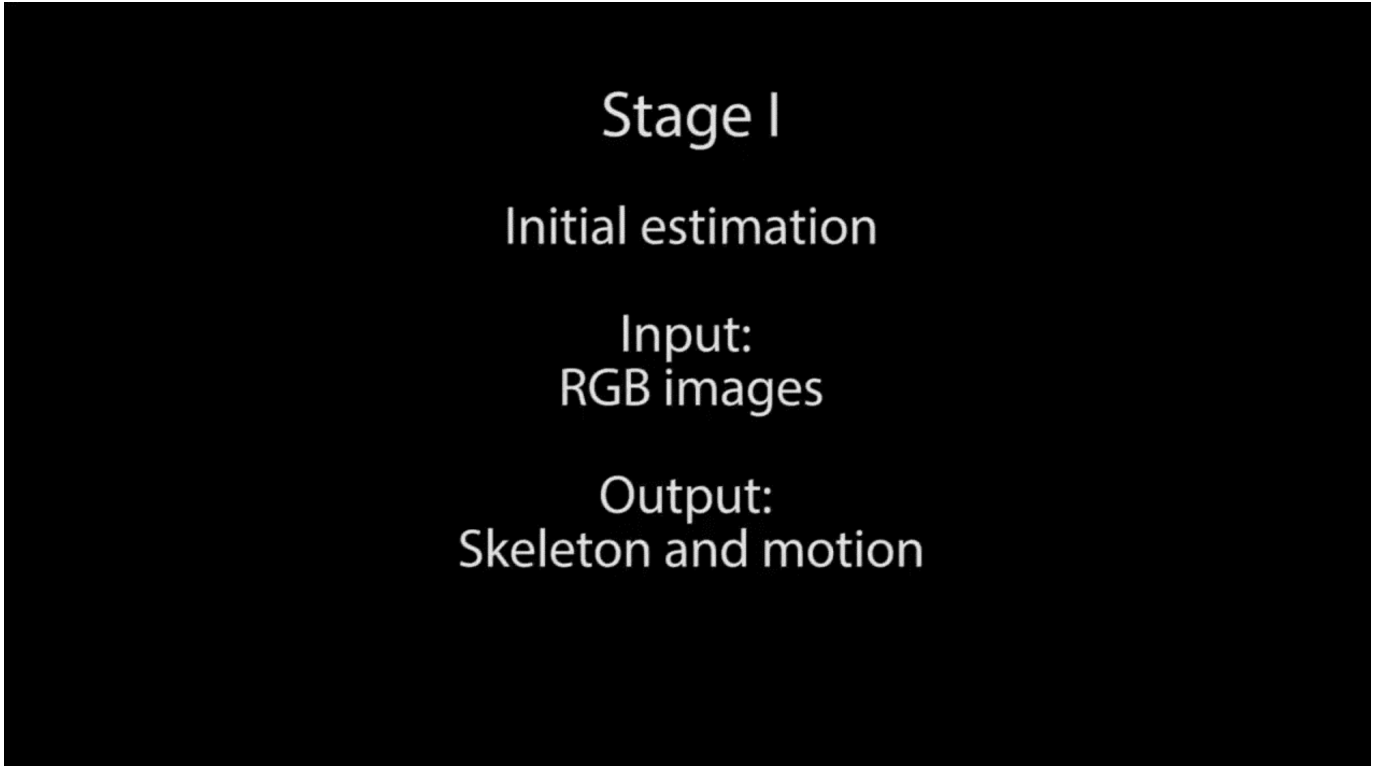
Multi-scale geometry and appearance model



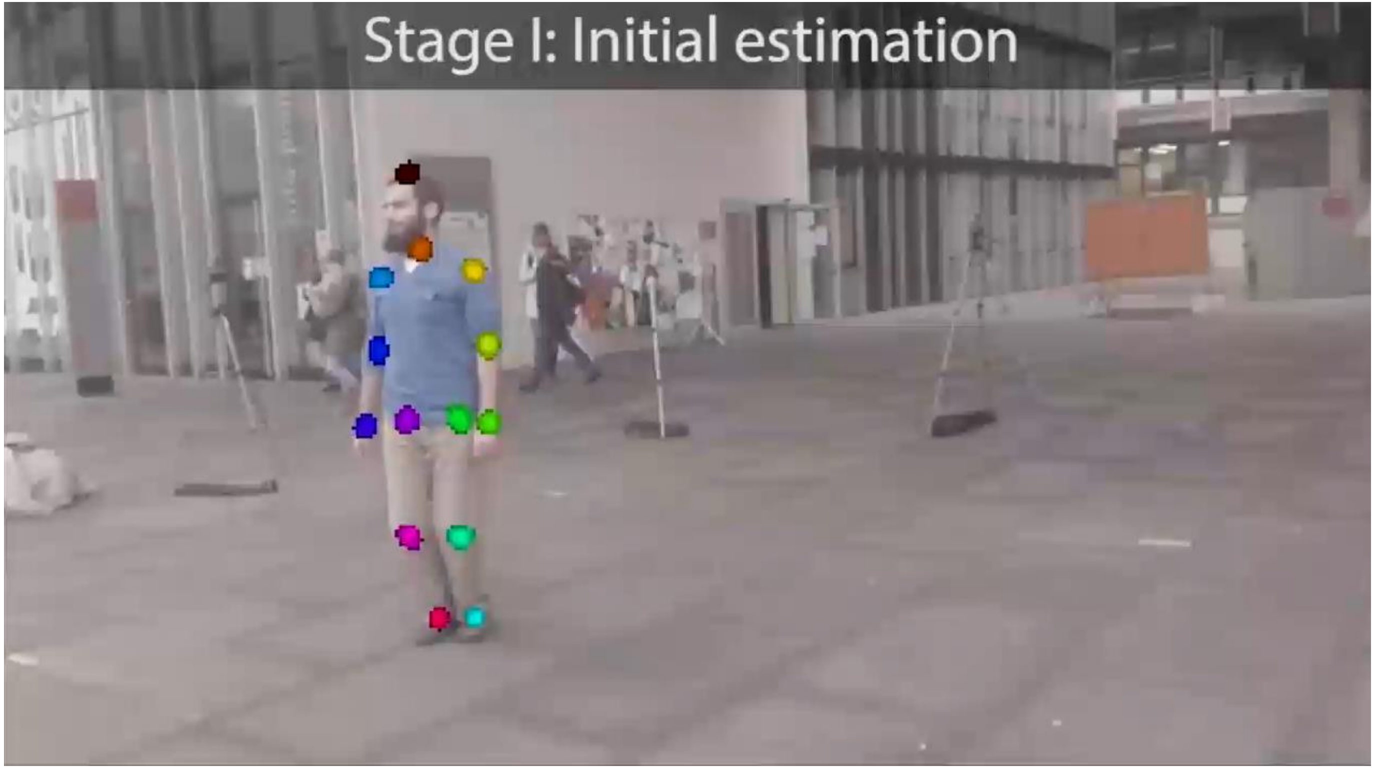
Parametric

Geometry + machine learning

Machine learning + Computer Graphics = Computer Vision



Machine learning + Computer Graphics = Computer Vision



Egocentric motion capture

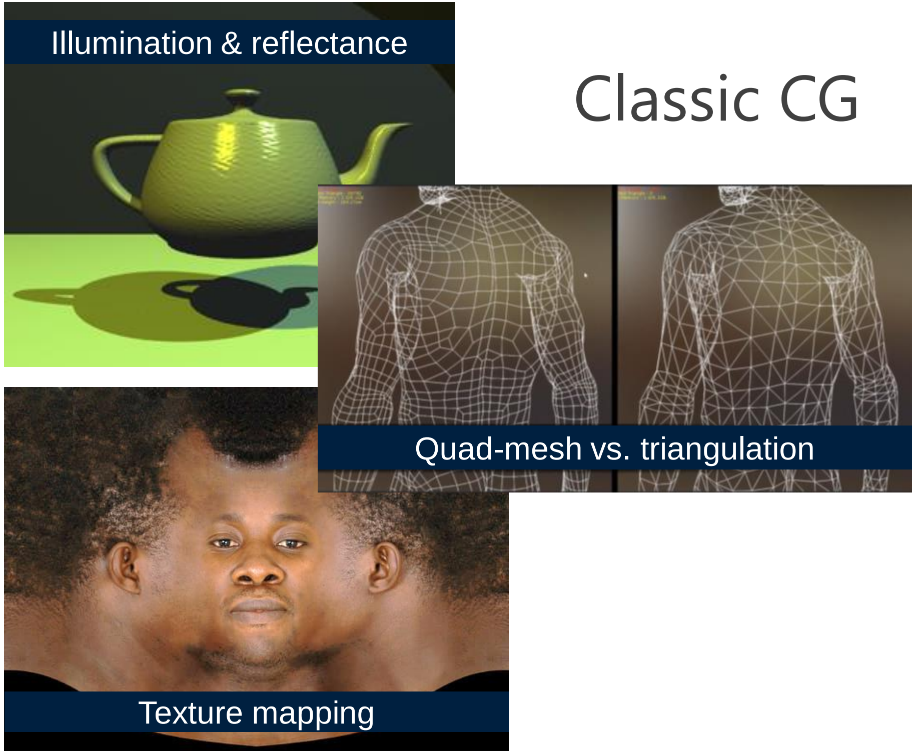
[EgoCap: Egocentric Marker-less Motion Capture with Two Fisheye Cameras]

Surface reconstruction

[MonoPerfCap: Human Performance Capture from Monocular Video, TOG 2018]

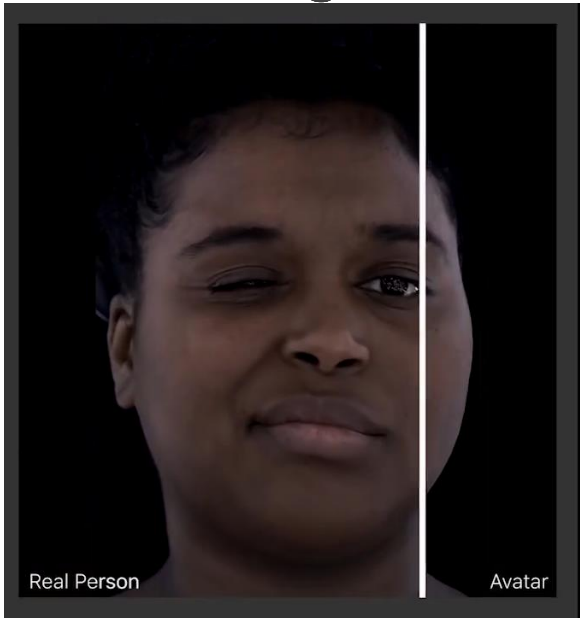
Computer Graphics

Classic CG



[Utah teapot, www.reallusion.com, www.turbosquid.com]

Learning-based CG



[Facebook AI Research]

Example topics

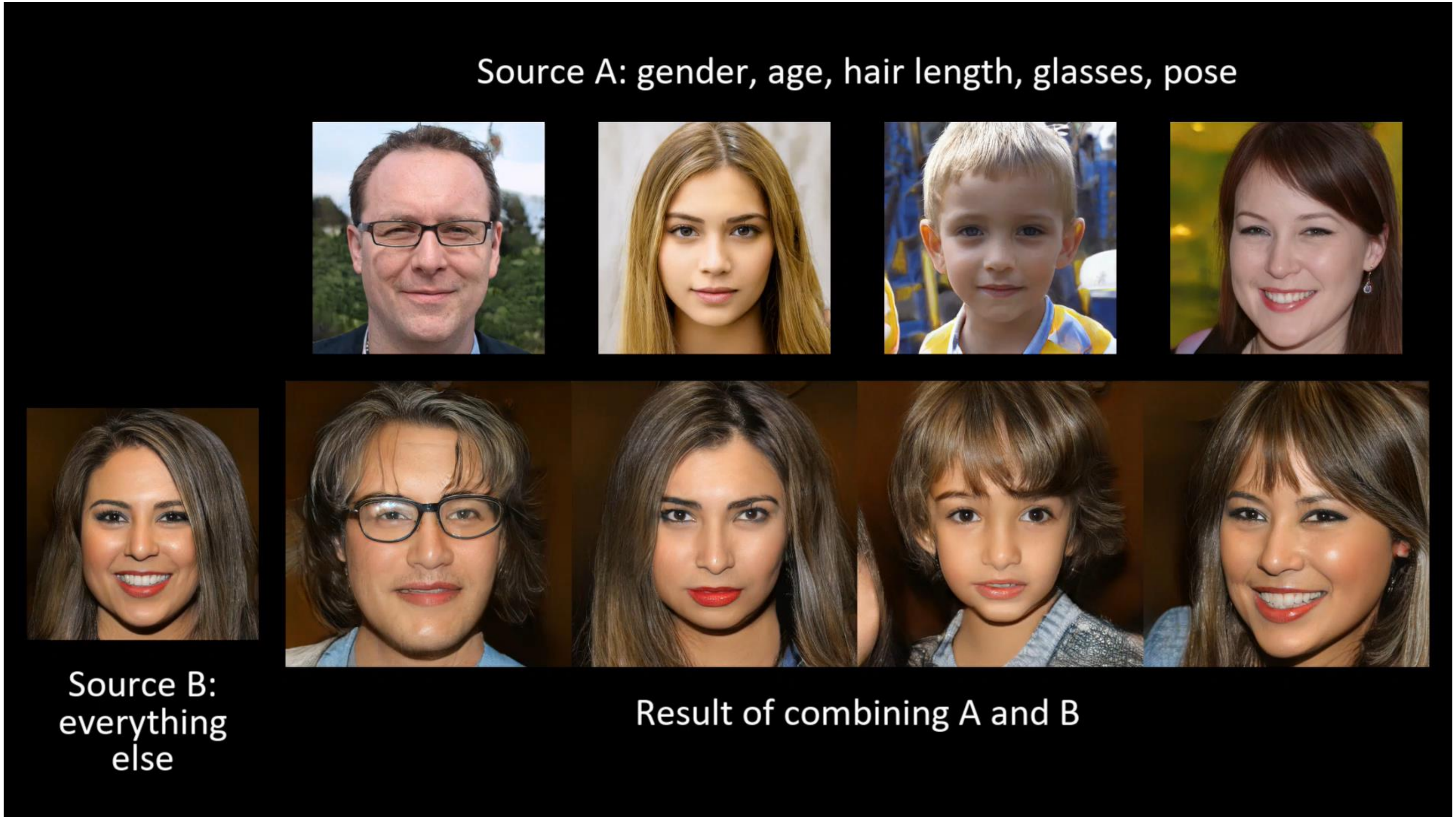
Everybody dance now, Chan et al. 2017

Computer Vision

Computer Graphics

100% Machine learning

StyleGAN – generating images from noise

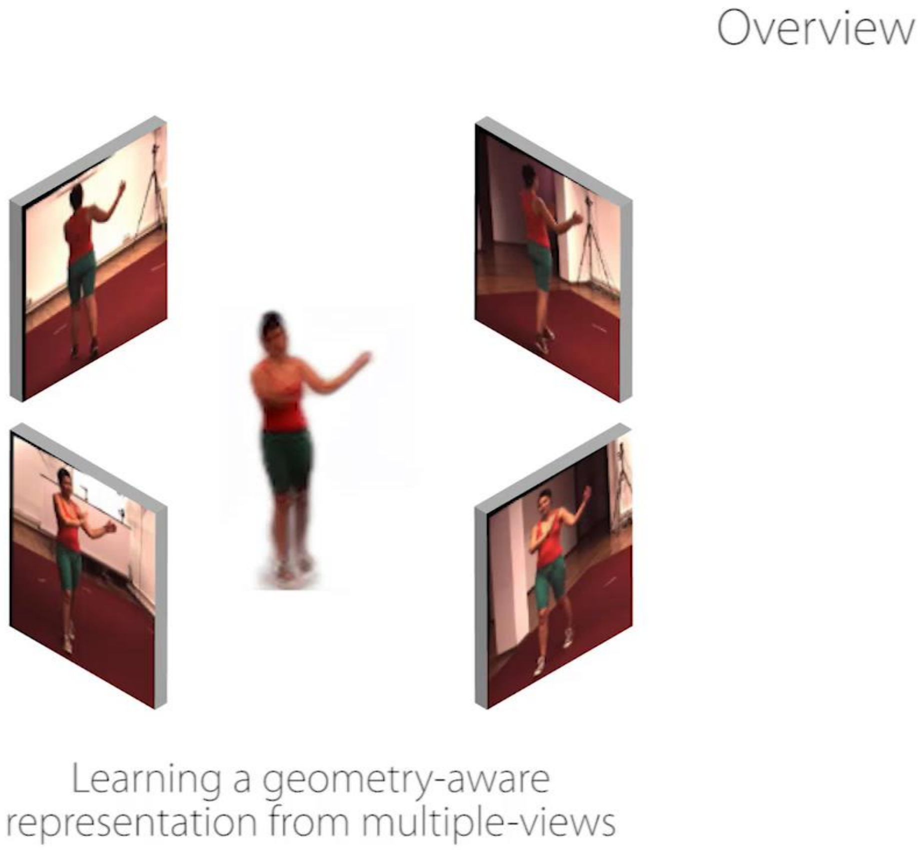


[StyleGAN: A Style-Based Generator Architecture for Generative Adversarial Networks. CVPR 2019]

HoloGAN – providing viewpoint control

[HoloGAN: Unsupervised learning of 3D representations from natural images, ICCV 2019]

Neural Humans



[Unsupervised Geometry-Aware Representation Learning for 3D Human Pose Estimation, ECCV 2018]

Disentangled appearance and pose

Appearance swap

Latent space interpolation

Image interpolation

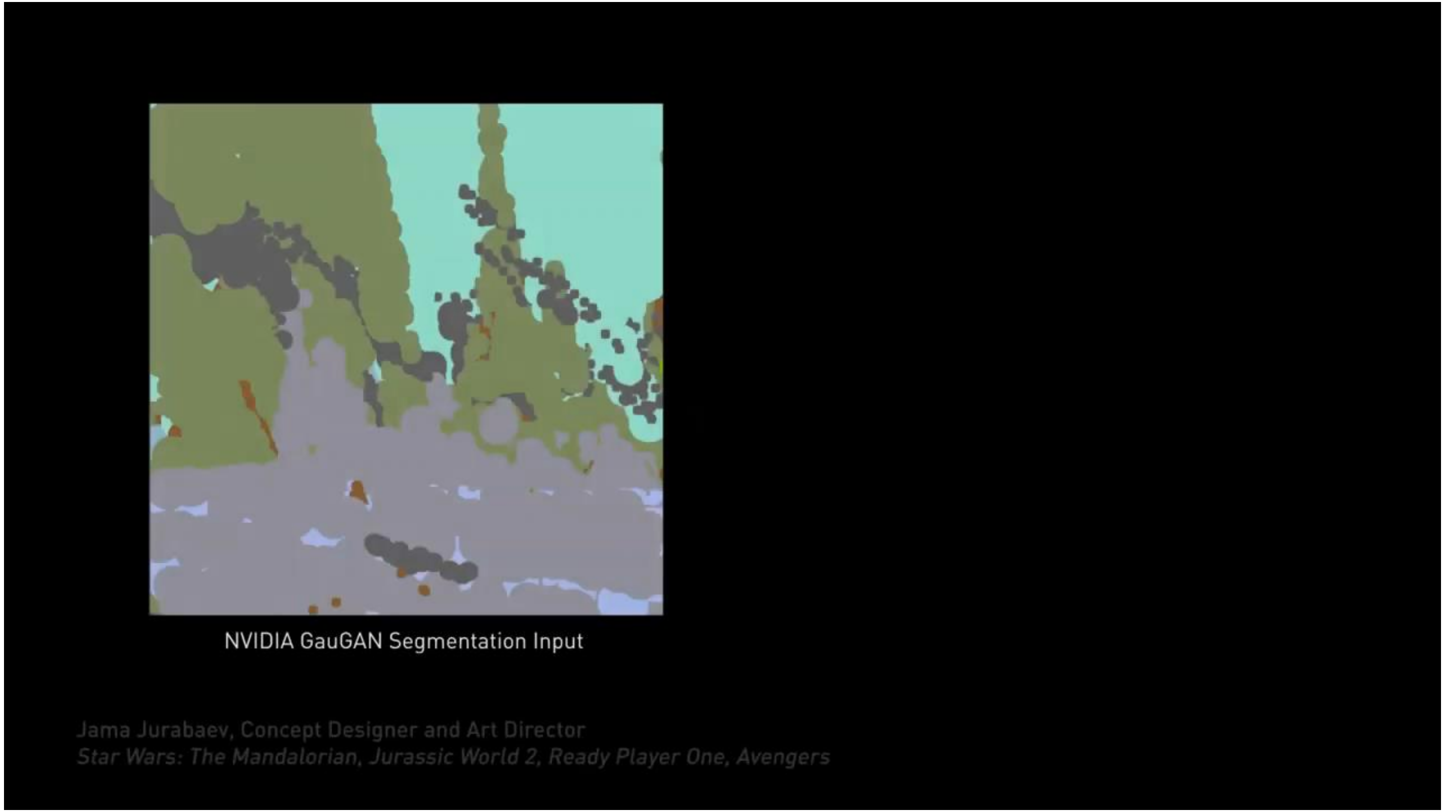
GAN Paint – conditional image generation

Online demo: http://ganpaint.io/demo/?project=church

[Semantic Photo Manipulation with a Generative Image Prior, SIGGRAPH 2019]

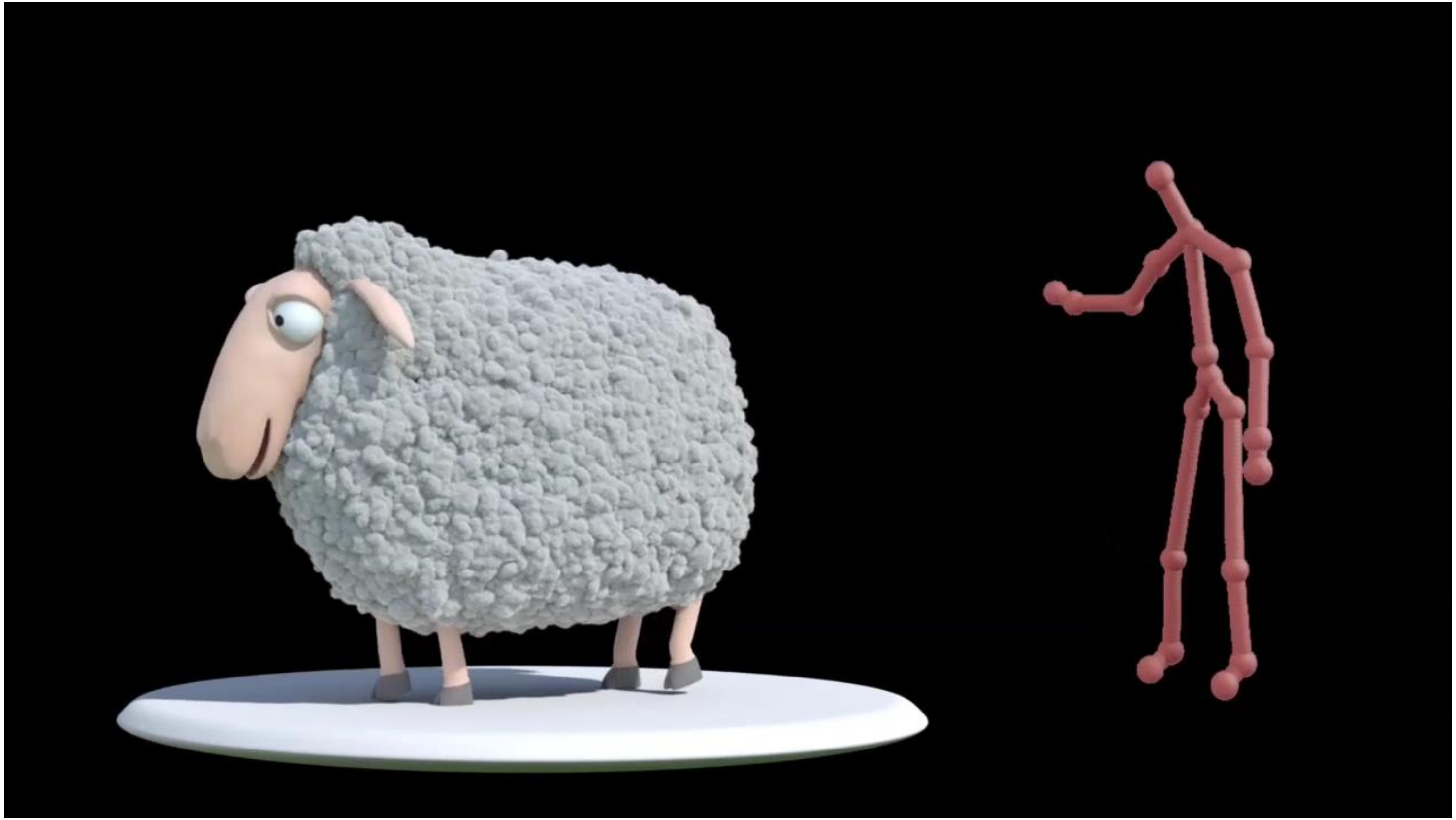
gauGan

Online demo http://nvidia-research-mingyuliu.com/gaugan



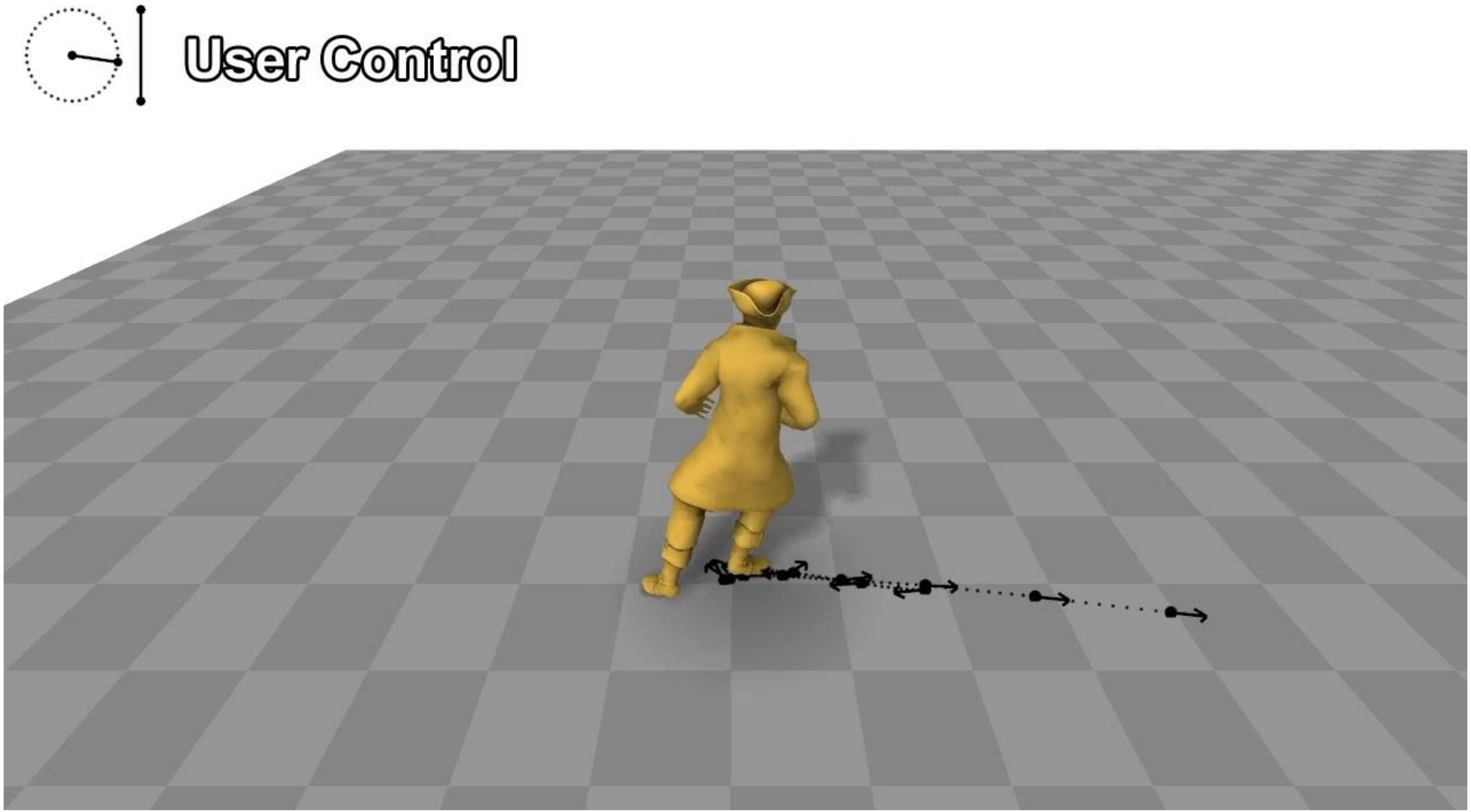
[Semantic Image Synthesis with Spatially-Adaptive Normalization, CVPR 2019]

Puppeteering



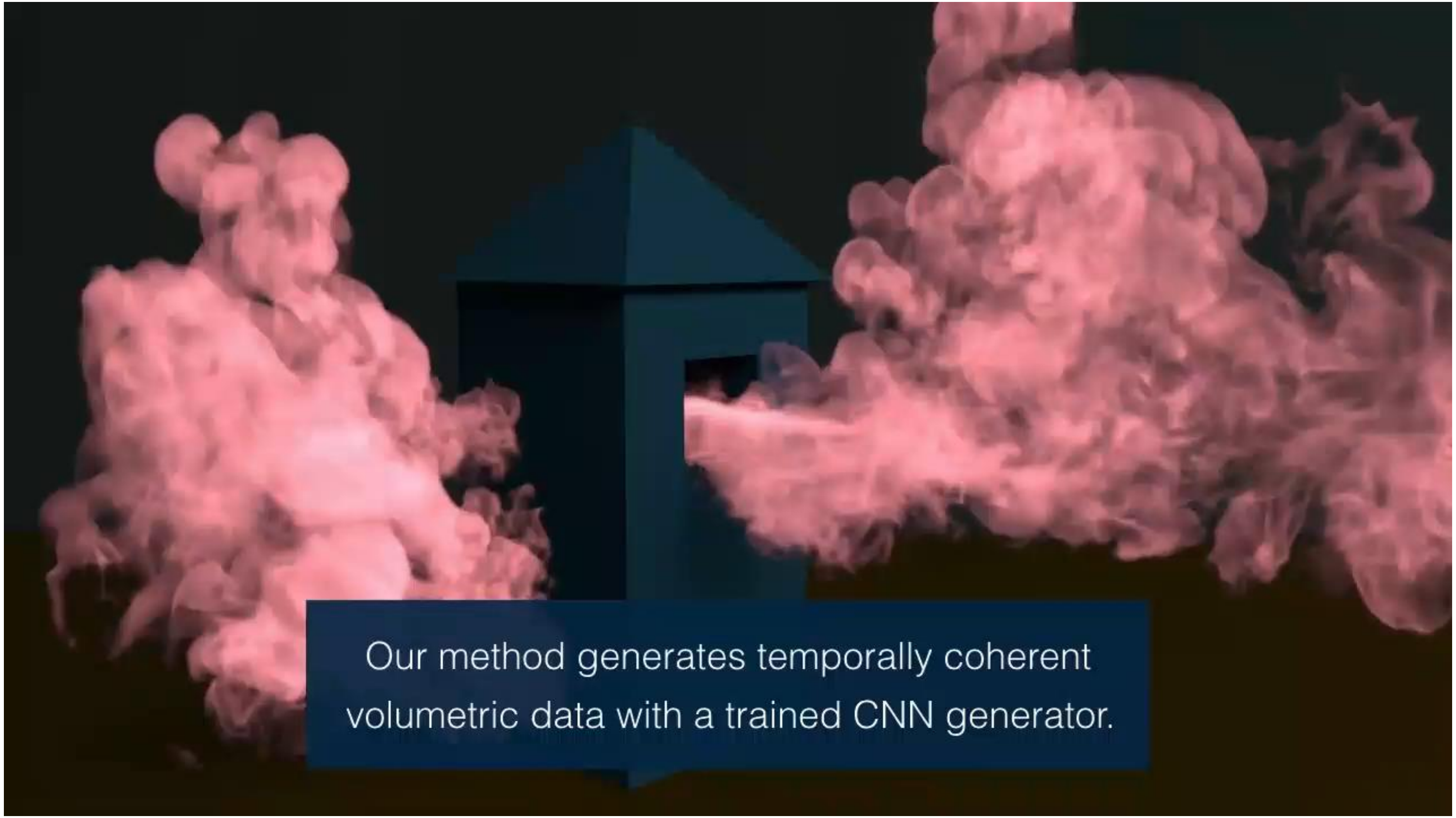
[Interactive Motion Mapping for Real-time Character Control, Eurographics 2014]

Character animation



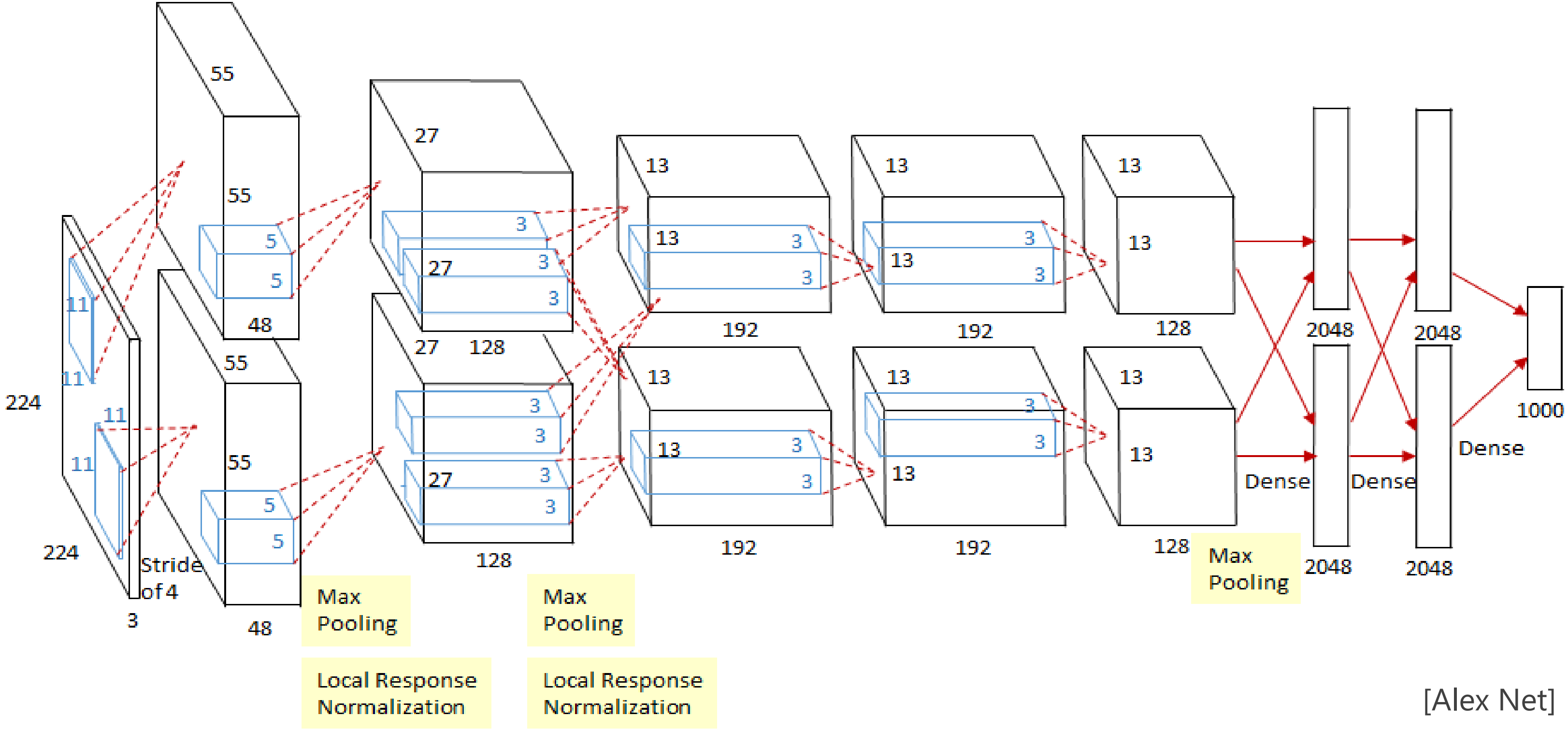
[Phase-Functioned Neural Networks for Character Control, SIGGRAPH 2017]

Physics simulation



[tempoGAN: A Temporally Coherent, Volumetric GAN for Super-resolution Fluid Flow, SIGGRAPH 2018]

Deep learning models = architecture?



[Alex Net]

Deep learning models for computer graphics and computer vision

Dataset creation

Optimization

Input, output and intermediate representations

Semi-supervised learning

Network architectures

Probability theory & distributions

Prime conferences

Graphics

- SIGGRAPH
- SIGGRAPH Asia
- Eurographics (EG)
- Journals: TOG and CFG

Computer vision

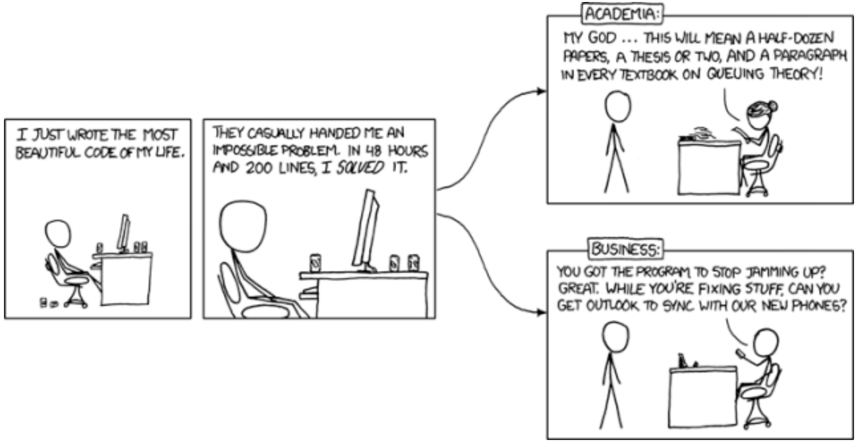
- Conference on Computer Vision and Pattern Recognition (CVPR)
- International Conference on Computer Vision (ICCV)
- European Conference on Computer Vision (ECCV) – every second year, alternates with ICCV
- Journals: IJCV and TPAMI

Machine learning

- Conference on Neural Information Processing (NeurIPS)
- International Conference on Machine Learning (ICML)
- International Conference on Learning Representations (ICLR)

Visual AI - Goals

- **Get to know and advance the state-of-the-art in Visual Computing**
  - Computer Graphics
  - Computer Vision
  - towards Visual AI
- Practice machine learning (ML)
  - From design through implementation to evaluation
  - Become a PyTorch and ML expert
    (PyTorch = deep learning framework)
  - tricks, hacks, gems, best practices, …
- Prepare you for academia (my group?!) and industry
  - **independently** complete a mini research project
  - become a researcher



https://www.explainxkcd.com/wiki/index.php/664:_Academia_vs._Business

Overview

- 11 Lectures (Weeks 1 – 6)
  - Introduction
  - Deep learning basics and best practices
  - Network architectures for image processing
  - Representing images and sparse 2D keypoints
  - Representing dense and 3D keypoints
  - Representing geometry and shape
  - Representation learning I (deterministic)
  - Representation learning II (probabilistic)
  - Sequential decision making
  - Unpaired image translation
  - Attention models

- 3x Assignments
  - Playing with pytorch (5% of points)
  - Pose estimation (10% of points)
  - Shape generation (10% of points)

- 1x Project (40 % of points)
  - Project pitch (3 min, week 6)
  - Project presentation (10 min, week 14)
  - Project report (8 pages, April 14)

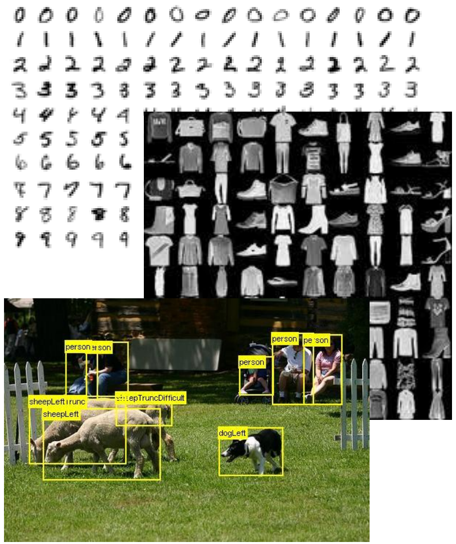
- 1x Paper presentation (Weeks 8 – 13)
  - Presentation, once per student (25% of points) (20 min + 15 min discussion, week 8-13)
  - Read and review one out of the two papers presented per session (10% of points)

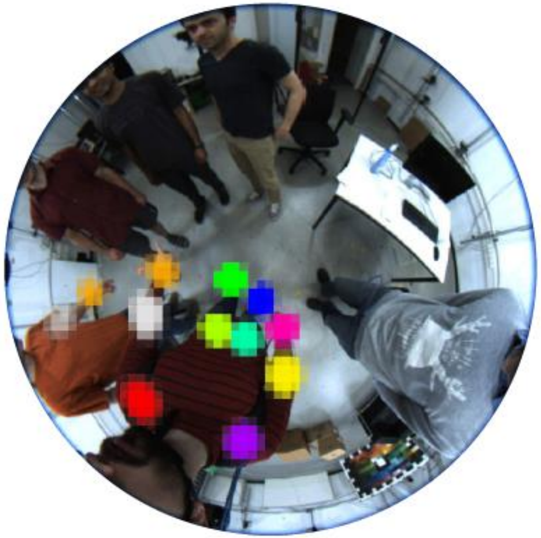
Assignments

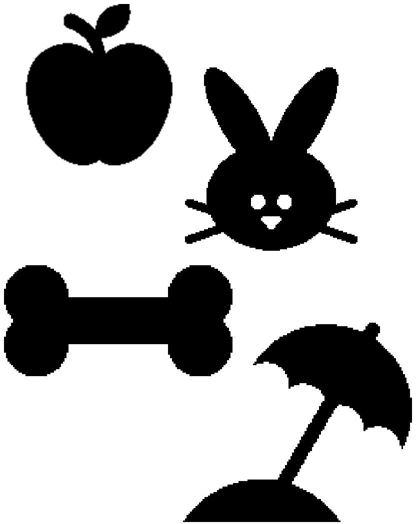
Assignment 1
Playing with PyTorch
(problem of your choice)

Assignment 2:
Egocentric human pose estimation
(2D human pose regression)

Assignment 3:
Shape generation
(variational auto encoder)







Assignments - rules

Academic Integrity. Assignments must be solved individually using the available course material and other online sources. You are neither allowed to copy nor look at parts of any of the assignments from anyone else. Accordingly, it is not allowed to post solutions online or distribute them in (private) forums. The university policies on academic integrity are rigorously applied.

Submission. Solutions must be handed in through the Canvas system (Not yet setup) and be kept private.

Deadline and grading. Assignments will be due on the dates specified in the schedule, always at 11:59 pm PST. A late submission by one day will still be accepted but reduces the score by -25%. The grading is based on correctness and completeness, as detailed in each exercise description.

Syllabus

- Talk assignment once all students are enrolled
  - Until then: open for paper suggestions
- Project topics choice before midterm break

Grading

Assignments (25%):

A1. Playing with PyTorch (5%)

A2. 3D human pose estimation, end-to-end training (10%)

A3. Learning shape models, unsupervised training (10%)

Presenting research (35%):

- Reviewing papers (10%)
- Presenting one paper (25%)

Project (40%)

- Proposal and experiment design
- Coding and evaluation
- LaTeX report
- Short project presentation

Lectures (preliminary schedule)

	Date	Content	Reading
W1	Jan 7	Introduction lecture slides - Challenges in using deep learning for creative tasks - Course expectations and grading - First steps in PyTorch Homework 1 posted assignment1.zip	SIGGRAPH program / video Pytorch intro
	Jan 9	Deep learning basics and best practices - regression/classification, objective functions - stochastic gradient descent, vanishing and exploding gradients.	Deep Learning Book - Chapter 8 Adam Optimizer
W2	Jan 14	Network architectures for image processing - Which neural network architectures work, why and how? - ResNet, DenseNet, UNet, FlowNet, MaskRCNN	Deep Learning Book - Chapter 9 ResNet, Unet
	Jan 16	Representing sparse 2D and 3D keypoints - heat maps, part-affinity fields - regression vs. classification Homework 1 due Homework 2 posted	Heat Maps Part Affinity Fields
W3	Jan 21	Representing dense and 3D keypoints - uv-coordinates, warp-fields - location maps, joint-angle skeleton	Dense Pose Location Maps
	Jan 23	Representing geometry and shape - voxels, implicit functions - PointNet, graph CNN, spiral convolution	PointNet Spiral convolution
W4	Jan 28	Representation learning I (deterministic) - principal component analysis (PCA) - auto-encoder (AE) Homework 2 due, Homework 3 posted	PCA face model Deep Learning Book - Chapter 14
	Jan 30	Representation learning II (probabilistic) - variational autoencoder (VAE) - generative adversarial network (GAN)	Deep Learning Book - Chapter 20
W5	Feb 4	Sequential decision making - Monte Carlo methods - reinforcement learning	Deep Learning Book - Chapter 17
	Feb 6	Unpaired image translation - cycle consistency - style transfer Homework 3 due	Cycle Gan Style transfer

W7		Midterm Break (no class)	-
W8	Feb 25	Conditional content generation Park et al., Semantic Image Synthesis with Spatially-Adaptive Normalization paper Li et al., Putting Humans in a Scene: Learning Affordance in 3D Indoor Environments paper	Read one of the two papers listed for each course. Submit review on the day before.
	Feb 27	Motion transfer Chan et al, Everybody Dance Now paper Gao et al., Automatic Unpaired Shape Deformation Transfer paper	
W9	March 3	Character animation Rhodin et al., Interactive Motion Mapping for Real-time Character Control paper Holden et al., Phase-Functioned Neural Networks for Character Control paper	
	March 5	Self-supervised learning Vondrick et al., Tracking Emerges by Colorizing Videos paper Doersch et al., Unsupervised visual representation learning by context prediction paper	
W10	March 10	Novel view synthesis Hinton et al., Transforming Auto-encoders, paper Rhodin et al., Geometry-Aware Representation Learning, paper	
	March 12	Differentiable rendering Rhodin et al., Differentiable Visibility, paper Wen. et al., TBA	
W11	March 17	Learning person models Lorenz et al., Unsupervised Part-Based Disentangling of Object Shape and Appearance paper Rhodin et al., Neural Scene Decomposition for Human Motion Capture paper	
	March 19	Object parts and physics Li et al., GRASS: Generative Recursive Autoencoders for Shape Structures, paper Xie et al., tempoGAN: A Temporally Coherent, Volumetric GAN for Super-resolution Fluid Flow paper	
W12	March 24	Objective functions and log-likelihood Jonathan T. Barron, A General and Adaptive Robust Loss Function paper Christopher Bishop, Mixture Density Networks paper	
	March 26	Self-supervised object detection Crawford et al., Spatially invariant unsupervised object detection with convolutional neural networks. paper Proposal optimization, TBA	
W13	March 31	Mesh processing Bagautdinov et al., Modeling Facial Geometry using Compositional VAEs paper Verma et al., Feastnet: Feature-steered graph convolutions for 3d shape analysis paper	
	April 2	Neural rendering Sitzmann et al., DeepVoxels: Learning Persistent 3D Feature Embeddings paper Saito et al., PIFu: Pixel-Aligned Implicit Function for High-Resolution Clothed Human Digitization paper	
W14	April 7	Project Presentations. (10 min talk per group, first half of groups)	-
	April 9	Project Presentations. (10 min talk per group, second half of groups)	-
	April 14	Project Report submission. (8 page PDF document, 11:59 pm)	-

3 min Break

Register on Piazza or play http://nvidia-research-mingyuliu.com/gaugan

Hidden questions

First steps in PyTorch

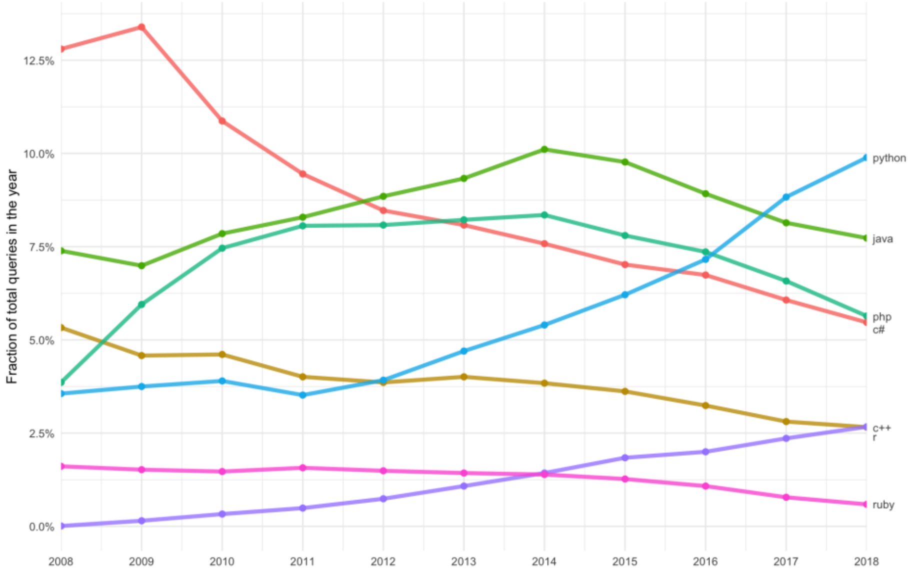
Programming environment - python™

Advantages

- high productivity / quick prototyping
- extensive support libraries
- high performance
  (with libraries linked to programs compiled from FORTRAN, C++, Cuda, …)

- we will use python 3!

Questions per year in Stack Overflow



https://towardsdatascience.com/predicting-the-future-popularity-of-programming-languages-4f28c80bd36f

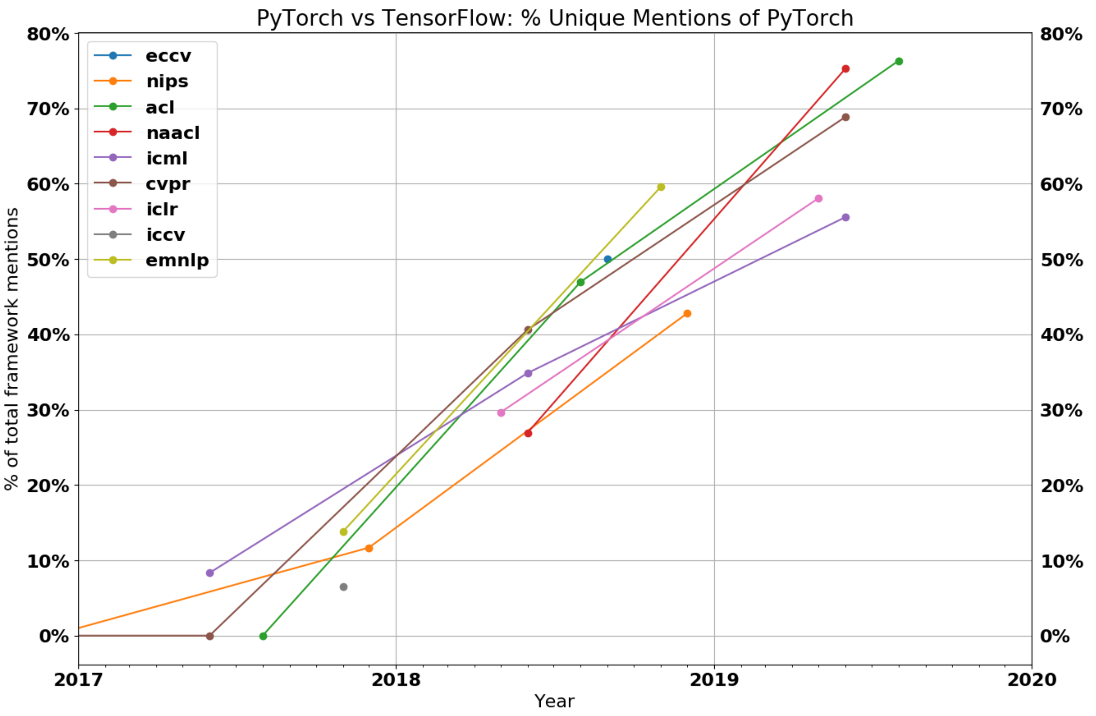
Machine learning framework - PYTORCH

Features

- efficient matrix and tensor operations (like NumPy)
- automatic differentiation (dynamic)
- large number of tutorials
- many open source repositories

Resources

- PyTorch tutorials https://pytorch.org/tutorials/
- PyTorch introduction https://pytorch.org/tutorials/beginner/deep_learning_60min_blitz.html



https://thegradient.pub/state-of-ml-frameworks-2019-pytorch-dominates-research-tensorflow-dominates-industry/

Version Control - git and GitHub

Version control system

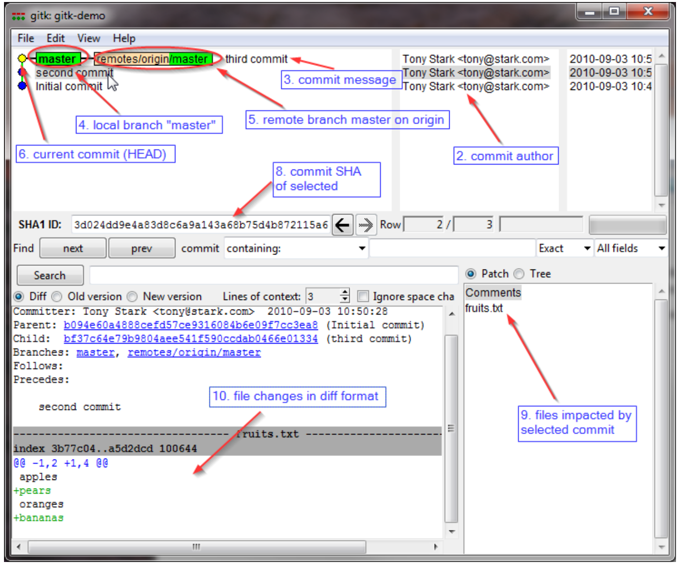
- Local repositories enable to track changes

```
git init
git add "Your_file.txt"
git commit –am "new commit"
```

- Remote repositories for backup and collaboration

```
git clone https://github.com/USER/REPO
git push origin master
```

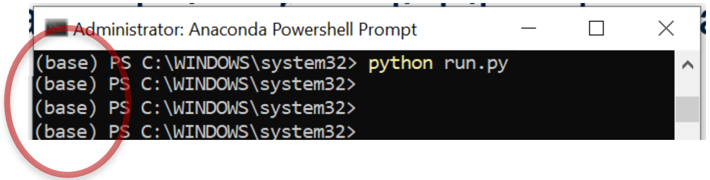
- Graphical version tree
  - `gitk`
  - More alternatives
    https://git-scm.com/download/gui/linux



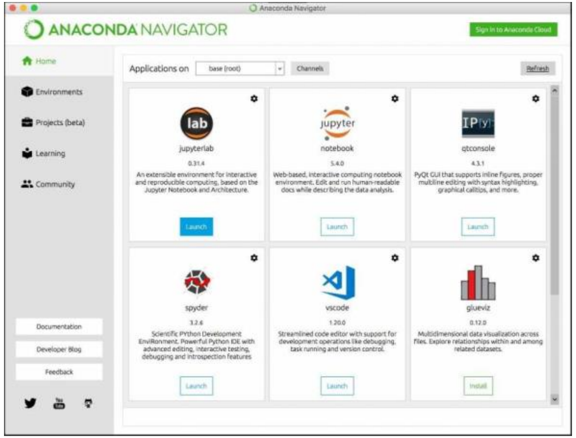
[https://lostechies.com/joshuaflanagan/2010/09/03/use-gitk-to-understand-git/]

Anaconda

- Anaconda is a free distribution of the Python programming languages
  - build for scientific computing
  - simplifies package management



- Virtual environment manager
  - conflicting packages can be installed independently
  - local installation without root access



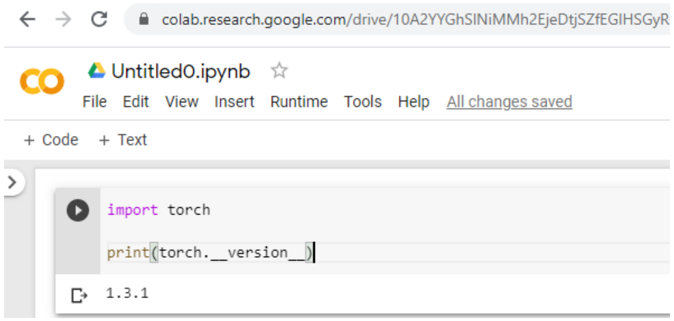
- Easy to use
  - graphical interface (Anaconda Navigator)
  - pre-compiled libraries

    “conda install pytorch torchvision cudatoolkit=10.1 -c pytorch”

Google colab

Cloud computing

- Provides a Jupyter notebook
- Incredible easy to setup
- Provides GPU access (for some time)
- Free of charge
- Interfaces with google drive



Deep learning and pytorch introduction

1. Neural network definition
2. Datasets, preprocessing, and efficient loading
3. Objective, optimization loop, and optimizer
4. Evaluation

Deep learning – a new way of programming

Classical programming

- Write down computational rules
  c = a + b
- Requires human programmer
  (domain expert + CS skills)

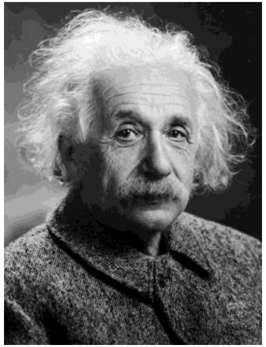
https://futurism.com/2-whats-the-next-blue-collar-job-coding

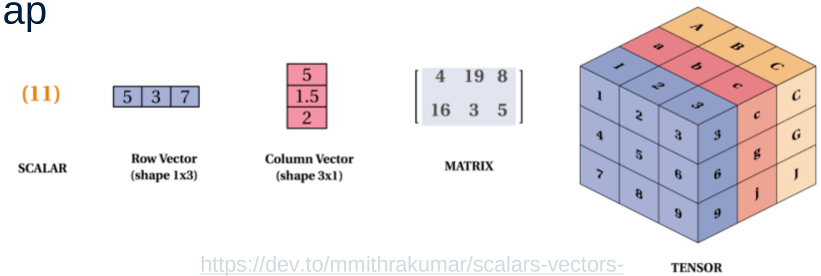
Data driven approach

- Give **lots of** input-output examples
  [(3,4) -> 7, (2,3) -> 5, (100,2) -> 102,
  (2,2) -> 4, (4,3) -> 7, …]
- Requires human annotator (domain expert)
- or Artificial Intelligence (AI) ?
  - Weak supervision, Self-supervision
  - Reinforcement learning …

Tensors in pytorch

- Tensor: a multi-dimensional array
  - scaler, vector, matrix, … tensor
- Term hijacked by ML community (in the math/physics community a tensor is a function that can be represented by a multi-dimensional array, but not every array is a math tensor)
- Pytorch uses the NCHW convention:
  dim 0: N, the number of images in a batch
  dim 1: C, the number of channels of an image / feature map
  dim 2: H, the height of the image / feature map
  dim 3: W, the width of the image / feature map
- Different #dimensions possible, dependent on the task





https://dev.to/mmithrakumar/scalars-vectors-matrices-and-tensors-with-tensorflow-2-0-1f66

- Order of dimensions matters (cache locality, parallelization)
  - TensorFlow has C in the last dimension, Nervada Neon N