Deep Self-Supervised and Semi-Supervised Learning

Christoph Bonitz

Twitter: @chris_bonitz
Linkedin: linkedin.com/in/christoph-bonitz

Goals

- Introduce semi-supervised and self-supervised learning
- Share some interesting success stories from the research community
- Distil common patterns

Outline

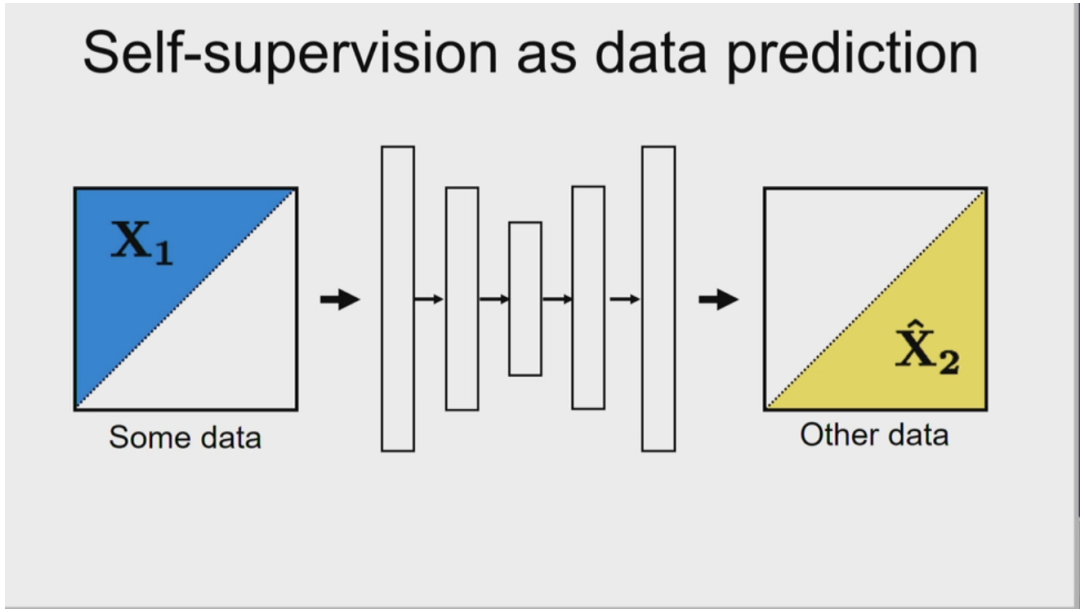
- Definitions
- Motivation
- Applications in NLP
- Applications in Vision
- Applications on Video
- Applications in Multi-Modal Representation Learning
- Conclusions

Definitions

Self-Supervised Learning

Andrew Zisserman defines it as

- *Form of unsupervised learning where the data provides the supervision*
- *In general, withhold some part of the data, and task the network with predicting it*
- *The task defines a proxy loss, and the network is forced to learn what we really care about, e.g. a semantic representation, in order to solve it*



Source: Alexei Efros The revolution will not be supervised, ICML 2019

Semi-Supervised Learning

Following Chapelle et al. (2006)

- Data is only partly supervised, i.e. there is data without targets.
- Other forms of partial supervision possible, e.g. these data points have the same target.

In practice, this means somehow learning the distribution of the data in an unsupervised way before training for the task.

Why Both in One Talk?

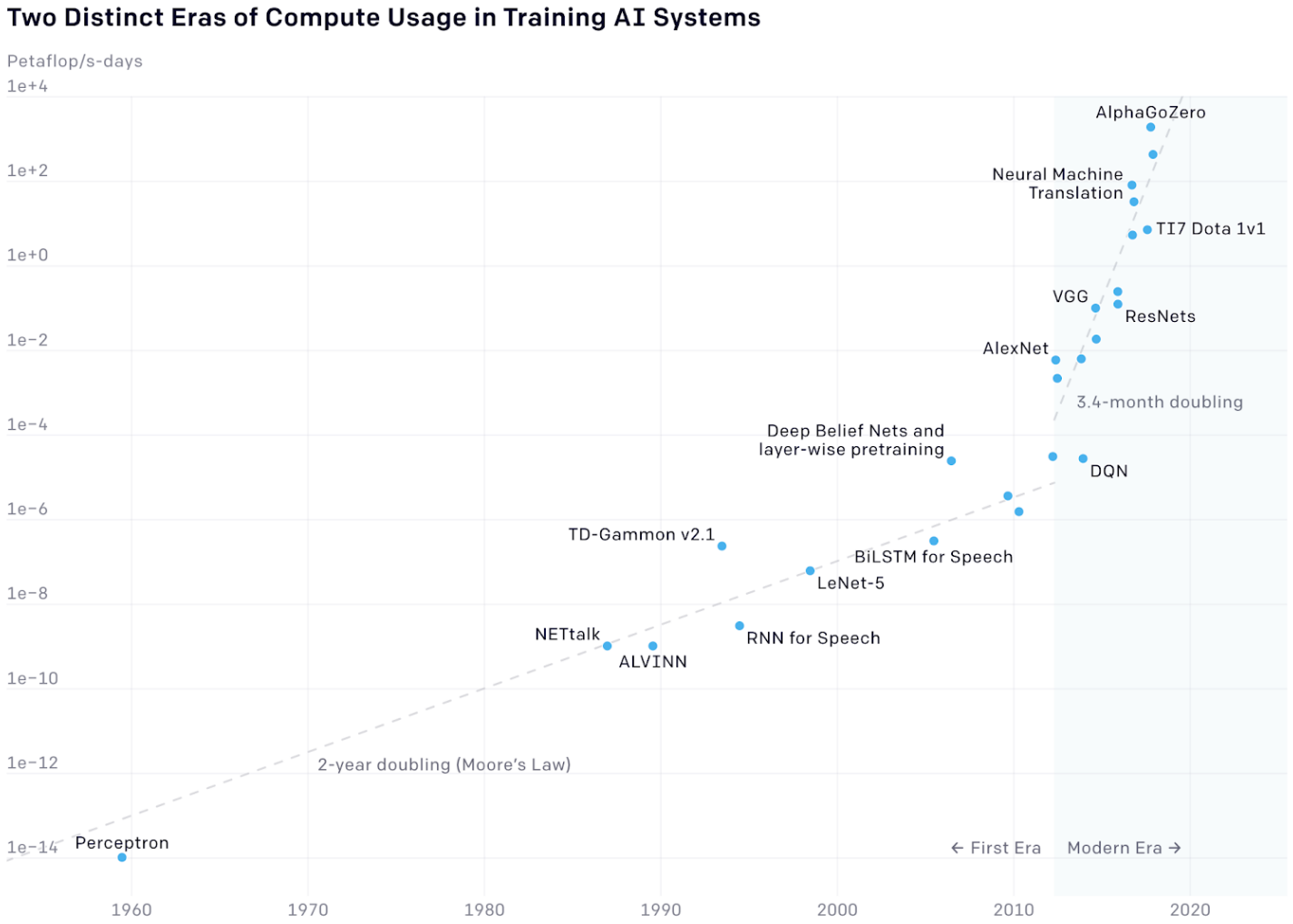
- Used together, e.g. self-supervised pretraining + supervised fine-tuning in BERT
- Common thread: Scale representation learning independently of human supervision. Why learn representations?
  - Information retrieval
  - Input for classification
  - Translation (literally or between different representations)
  - Anomaly detection
  - Features for downstream tasks

Why This Talk?

- Read and heard more and more about research in this area recently
- Find it fascinating for two reasons
  - Some very smart and original ideas about how to extract a learning signal from unsupervised data.
  - In some areas, results that feel like magic.
- Value I want to provide: Curate research that is interesting enough to make you *want* to learn more about the topic.

Motivation

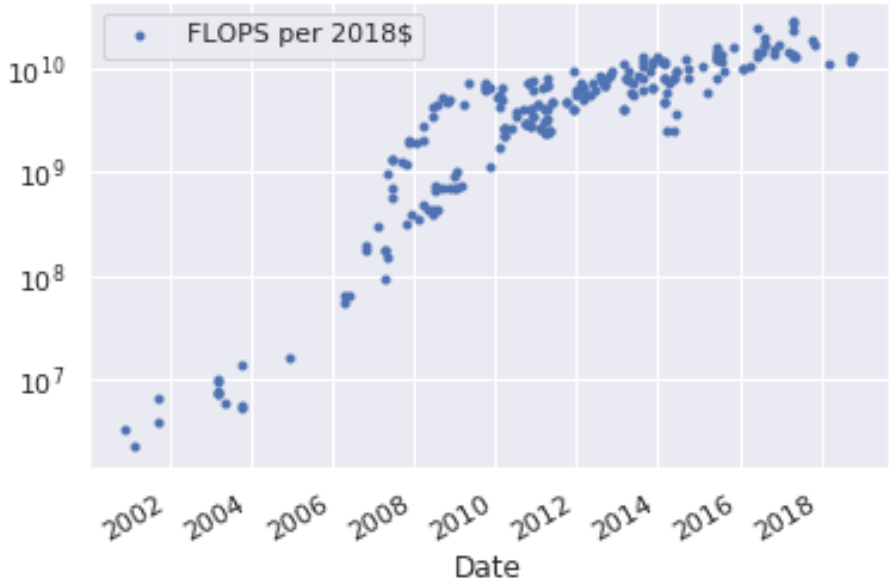
Amount of Compute That Can Be Utilized for ML Models Is Increasing Rapidly



[Source: OpenAI Blog]

Compute Is Getting Cheaper

- FLOPS per (non-inflation adjusted) USD paid for a GPU at launch price



Source: Median Group

Human-Labelled Data Is Expensive

- You don’t always have to directly pay for labelling data.
  - Some applications produce labelled data as a by-product.
  - Some applications are built exclusively to generate training data.
- If you have to, though, there are almost no economies of scale.
- Example: Breakthrough Ophtalmology DL paper De Fauw et al., *Clinically applicable deep learning for diagnosis and referral in retinal disease. Nature Medicine.* Nat Med 24, 1342–1350 (2018) (pdf)

	Dataset	Device type	Number of scans	Input	Labels	Label source
#1	Training set for segmentation	1	877	OCT scans	Sparse segm. maps (3-5 slices per scan)	Manually segmented by trained ophthalmologists, reviewed and edited by a senior ophthalmologist.

Human-Labelled Data Is Not Always Available

- Example: Schlegl, T. et al. *f-AnoGAN: Fast Unsupervised Anomaly Detection with Generative Adversarial Networks*. Medical Image Analysis 54, 30-44. (GitHub)
  - If your goal is biomarker discovery in medical images, you cannot create labeled data for the the unknown biomarker.

Oversimplified Practicalities of Learning

	Building and training the model	Supervised data required	Data leveraged
Supervised learning from scratch	Well-understood problem	A lot	Supervised dataset
Transfer Learning	Well-understood problem	Very little	Pre-training dataset + supervised dataset
Self-supervised pre-training	Research problem	Very little	Usupervised pre-training dataset + supervised dataset
Self-supervised / Unsupervised	Research problem	None	Unsupervised dataset

Oversimplified Economics of Learning

	Model development	Creating initial dataset	Scaling the dataset
Supervised learning from scratch	cheap	expensive because big	expensive
Transfer Learning	cheap	cheap because small	expensive
Self-supervised pre-training	expensive, risky	pretraining: cheap because unsupervised fine-tuning: cheap because small	pretraining: cheap fine-tuning: expensive
Self-supervised / Unsupervised	expensive, risky	cheap because unsupervised	cheap

1st Conclusion

- Being able to not (only) rely on human-labelled data can help take advantage of some very beneficial macro-trends and can have very attractive economics.

But Wait, I've Heard That Before…

- Just before the golden era of Deep Learning, unsupervised methods were already very popular and considered by some a necessary part of ML training
  - Example: Deep Belief Networks to learn initial weights for NN training.

What is wrong with back-propagation?

- It requires labeled training data.
  - Almost all data is unlabeled.
- The learning time does not scale well
  - It is very slow in networks with multiple hidden layers.
- It can get stuck in poor local optima.

Source: G. Hinton, NIPS tutorial on DBNs (2007)

Why Did the 2010s Take a Different Turn?

- Several fundamental techniques were discovered in rapid succession:
  - Better gradient descent variants, e.g. Adam
  - Better non-linearities, e.g. ReLU
  - Better random initialization, e.g. He et al.
  - Skip-connections
  - ….
- Together, they enabled training of deep neural networks from scratch.
- Some very large supervised datasets, e.g. ImageNet
- Transfer learning
- Data augmentation techniques help make better use of limited data.

What's Different Now? (My Take)

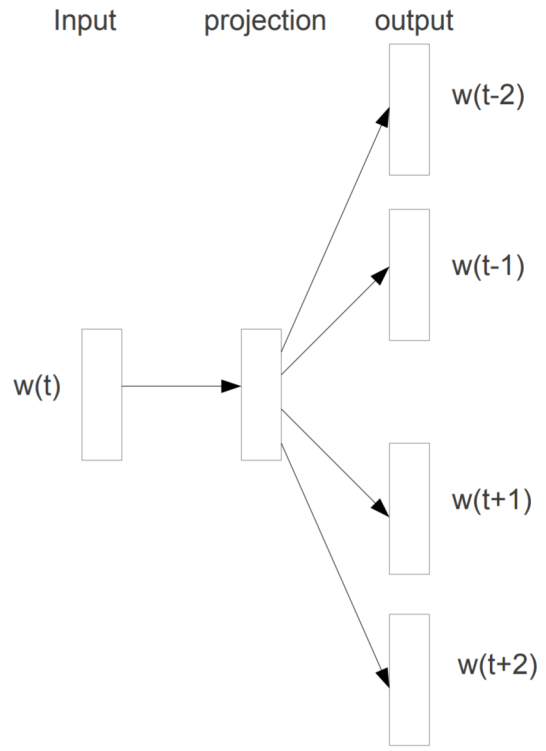
- Real-life problems with higher-dimensional output
  - Classification < object detection < image segmentation < video segmentation
- More building blocks discovered and refined
  - Skip-grams
  - Masking
  - Transformers
  - GANs
  - …
- Success stories in NLP, image processing and cross-modal retrieval indicate research community is starting to master these techniques.

Applications in Natural Language Processing

Early Success: Word2Vec

[Mikolov et al., Distributed Representations of Words and Phrases and their Compositionality, NIPS (2013)]

- Basic principle: Try to guess the context of a word (words before and after it) *Skip-Gram*
  - Classic self-supervision.
  - Uses shallow network.
- Turns one-hot into vector-representations, commonly used as input features for other language models
- Can do arithmetic with vectors (king - man + woman)
- Related: GloVe (Pennington et al. 2014), based on co-occurrence matrix



Source: Mikolov et al. 2013

Paradigm Shift: Fine-Tuning of Pre-Trained Transformer-Type Networks

Devlin et al., BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv preprint (2018)

- BERT, ELMo, ULMFiT, GPT etc: Self-supervised on large corpora
- BERT trained by
  - Masking words in text
  - Next sentence prediction
- Generate useful representations that can be fine-tuned for multiple NLP tasks
- Self-supervised pre-training, supervised fine tuning => semi-supervised end-to-end. Now leveraging large corpora even for tasks with limited data.

Patterns

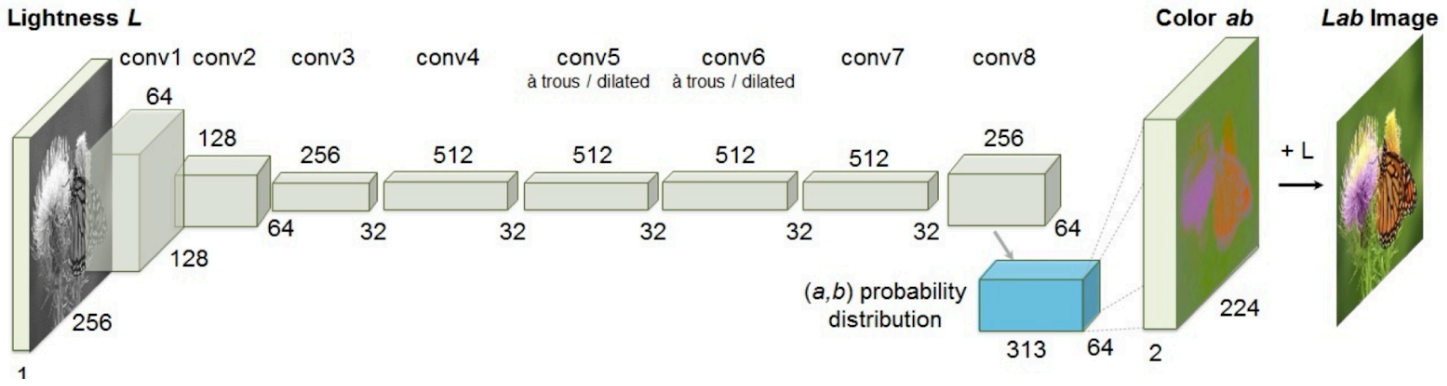
- Remove words from your text to create a training signal

Applications in Vision

Colourising Black and White Images

Zhang et al., *Colorful Image Colorization*, ECCV (2016)

- Separate LAB colour space into brightness and colour, let the network find a colorization
- Instead of predicting a colour, output a colour distribution and use annealing to create colourisation – avoids lots of brownish areas.



Source: Zhang et al. 2016

Examples:
Various colourised images
Colourised Henri Cartier-Bresson

Try it yourself

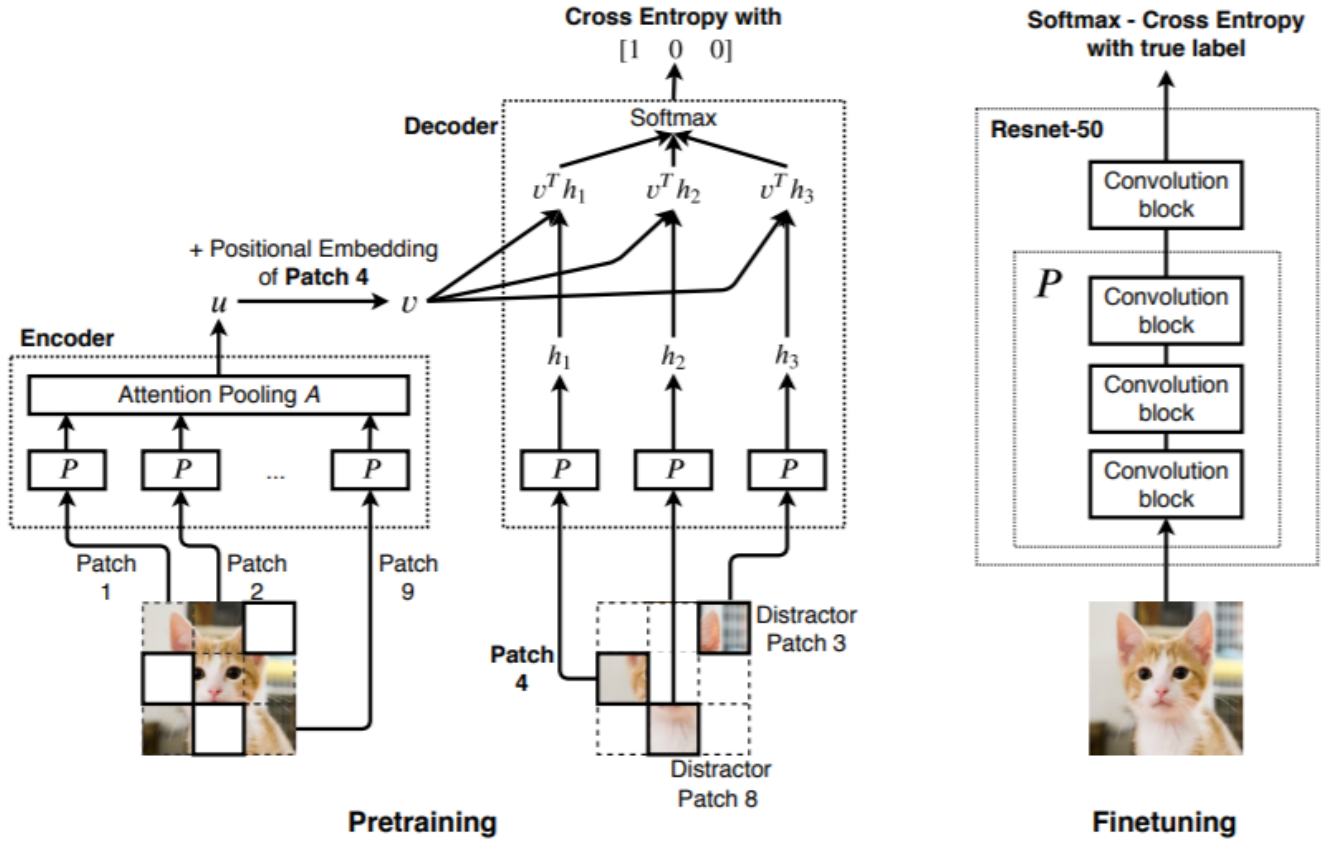
Pre-Training ConvNets

Trinh et al., *Selfie: Self-supervised Pretraining for Image Embedding (2019)*

- Pre-train ConvNet by using representations to fill in missing patches of an image.
  - Create representation of the image.
  - Using only that, answer which of several patches goes to a particular spot.
  - True patch and distractor patches,
  - Use cross-entropy to train
- Supervised fine-tuning for classification

Pre-Training ConvNets

Trinh et al., *Selfie: Self-supervised Pretraining for Image Embedding (2019)*



Source: Trinh et al. 2019

Pre-Training ConvNets

Trinh et al., *Selfie: Self-supervised Pretraining for Image Embedding (2019)*

- Better and more stable results than supervised training on same amount of labelled examples. More improvement with fewer examples.
- Researchers theorise that to solve the training problem, the network has to
  - Understand the content of the whole image
  - The local content of each patch
  - Their relationship

Monocular Depth Estimation

- Depth estimation important for many applications, e.g.
  - Robotics
  - Virtual and Augmented Reality
  - Photo processing
    - Changing the lighting
    - Simulating lens blur
    - Adding synthetic objects
- LIDARs solve that problem very well but are large and expensive
- Classical computer stereo vision methods work well with calibrated stereo cameras for static scenes
- Deep learning used more and more as part of pipelines or end-to-end.
- Holy grail: Robust monocular depth estimation from cheap cameras

Monocular Depth Estimation

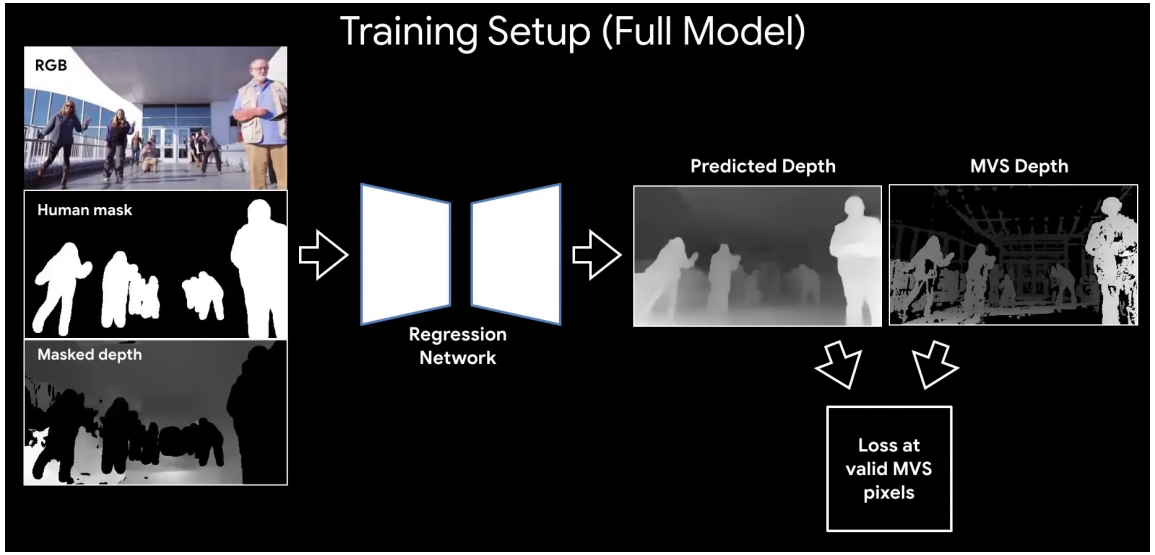
Zhengqi Li et al., Learning the Depths of Moving People by Watching Frozen People CVPR, (2019) (video)

- Could use geometry-based methods to cheaply create ground truth but they don't work well when the scene changes between images.
- Scenes with people change a lot, but we would like depth estimation to work well on those.
- Enter the Mannequin Challenge – people acting as if they were frozen in time, with camera moving through the scene
  - Static scene, but with people
  - Camera movement enabling Multi View Stereo methods for ground truth
  - Human supervision limited to choosing the right videos as input

Monocular Depth Estimation

[Zhengqi Li et al., Learning the Depths of Moving People by Watching Frozen People CVPR, (2019)](video) (video)

- Hourglass-type architecture, inputs that can be computed by existing methods



Source: Supplementary video (Tali Tekel)

"... impaints and refines... depth map"

Patterns

- Remove words from your text to create a training signal
- Remove spatial or dimensional parts of your image to create a training signal
- Output a distribution instead of a value to avoid boring results
- Human supervision = selecting the right data to train on
- Use computed data or other sensors as training signal

Multi-Modal Representation Learning

Multi-Modal Representation Learning

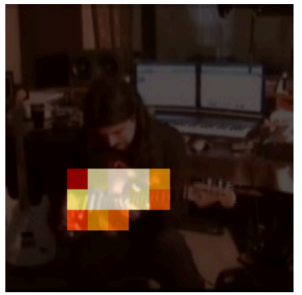
- We perceive the world with many senses, and easily correlate different representations of the same thing.
- This is a difficult problem for information retrieval systems. Learning robust vector-valued representations that work across modalities would be very helpful.
- Correspondences between different modalities in data that is inherently multi-modal (e.g. Video) can provide a learning signal.
- The following papers do that, and leverage the representations for very unconventional retrieval tasks.

Where Does That Sound Come From?

[Arandejelovic R. et al. *Objects that Sound* (2018)]

- Show which part of the image is creating a sound.
- Training only uses *audio-visual correspondence*, i.e. let the network decide whether a video frame and one second of audio correspond or not.

(a) Input image with sound



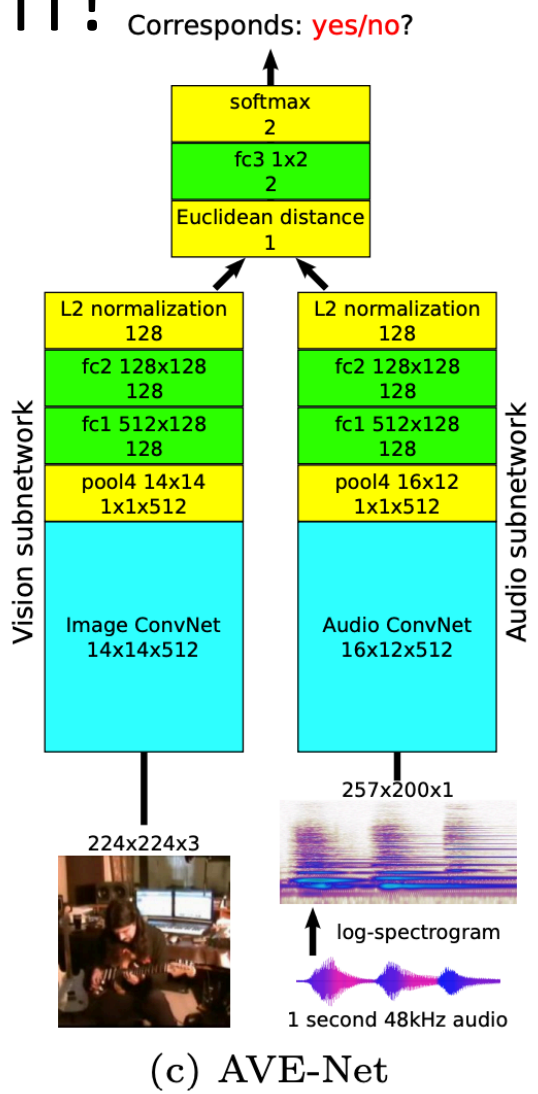
(b) Where is the sound?

Source: Arandejelovic et al. 2018

Where Does That Sound Come From?

[Arandejelovic R. et al. *Objects that Sound* (2018)]

- Two ConvNets compute 128-vector representation, Euclidian distance between vectors goes into a SoftMax for classification



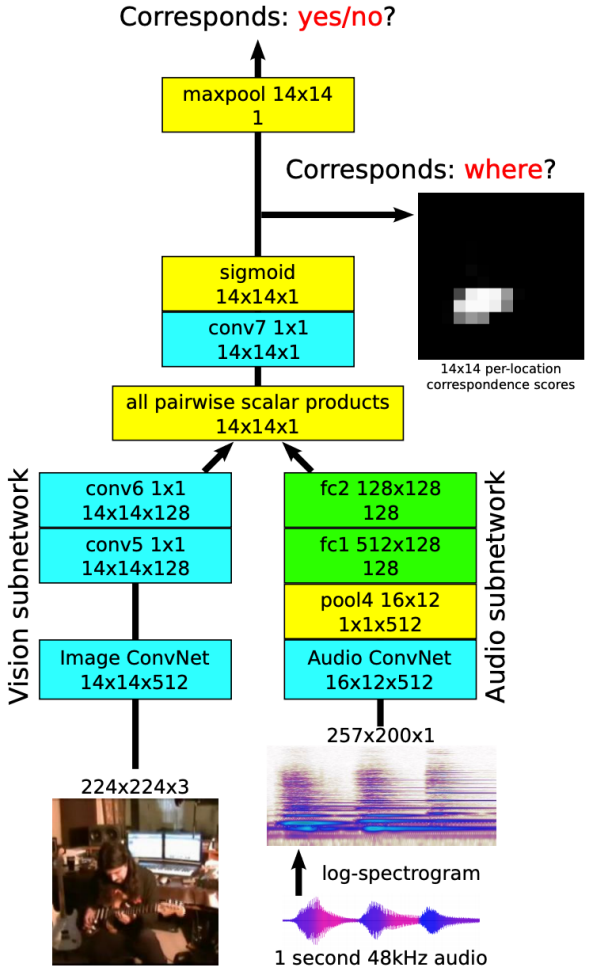
(c) AVE-Net

Source: Arandejelovic et al. 2018

Where Does That Sound Come From?

[Arandejelovic R. et al. *Objects that Sound* (2018)]

- How to find the location? Keep spatial structure and use a heatmap instead of distance between vector representations and use maxpool layer for discrimination.
- Still only training on correspondence.



Source: Arandejelovic et al. 2018

Interlude: Beware of Trivial Shortcuts

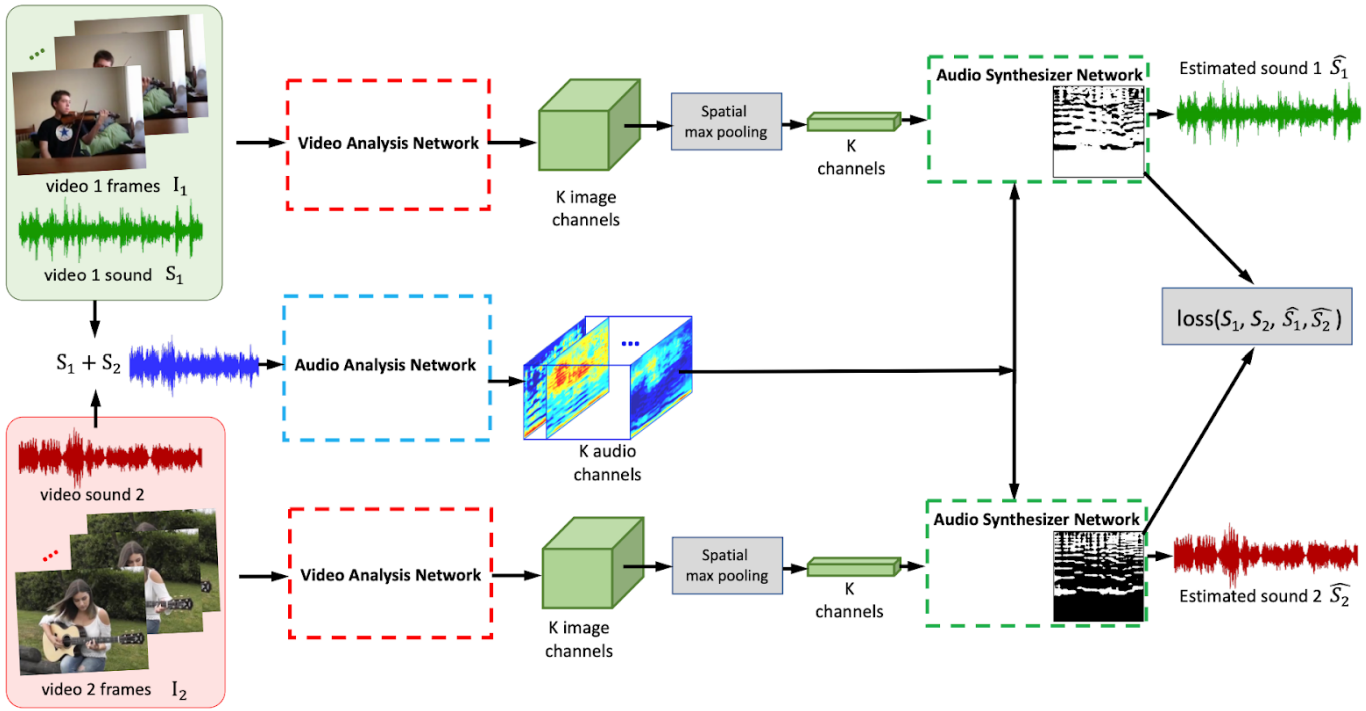
Arandejelovic R. et al. *Objects that Sound* (2018)

- Cautionary tale: Initially, positive sound samples were perfectly aligned around the corresponding frame, whereas negative ones were randomly chosen.
- Network learned to recognise sound samples that were perfectly centered around a frame vs ones that weren't, probably due to some low level encoding artifact.
- In Doersch et al., *Unsupervised visual representation learning by context prediction.,* Proc. CVPR. (2015), a network learned to place image patches via chromatic aberration, a lens artifact.

How Does That Part of the Image Sound?

Hang Zhao et al., *The Sound of Pixels*, ECCV 2018 (Website)

Training: make network separate artificially mixed audio sources based on source video (*Mix-and-Separate*)

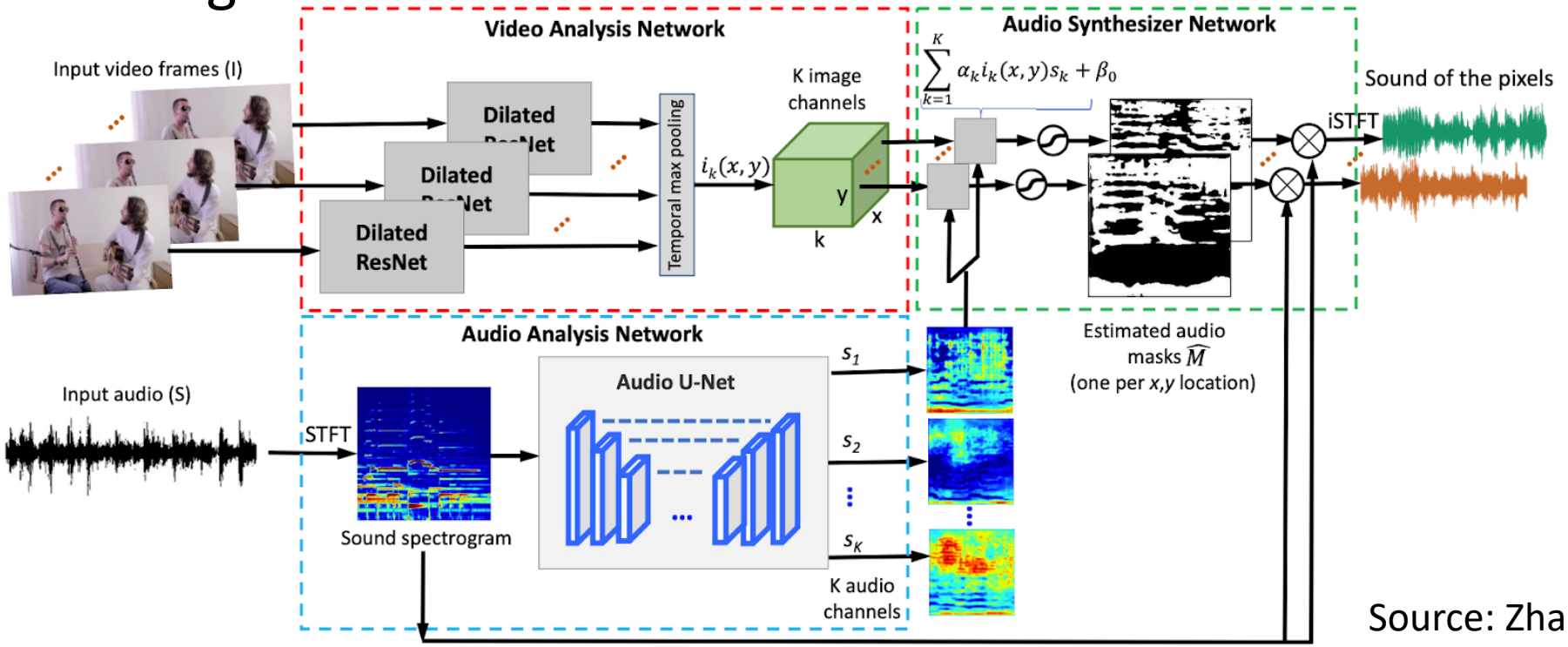


Source: Zhao et al. 2018

How Does That Part of the Image Sound?

Hang Zhao et al., *The Sound of Pixels*, ECCV 2018 (Website)

Inference: Use non-pooled, local image representation to retrieve matching audio.

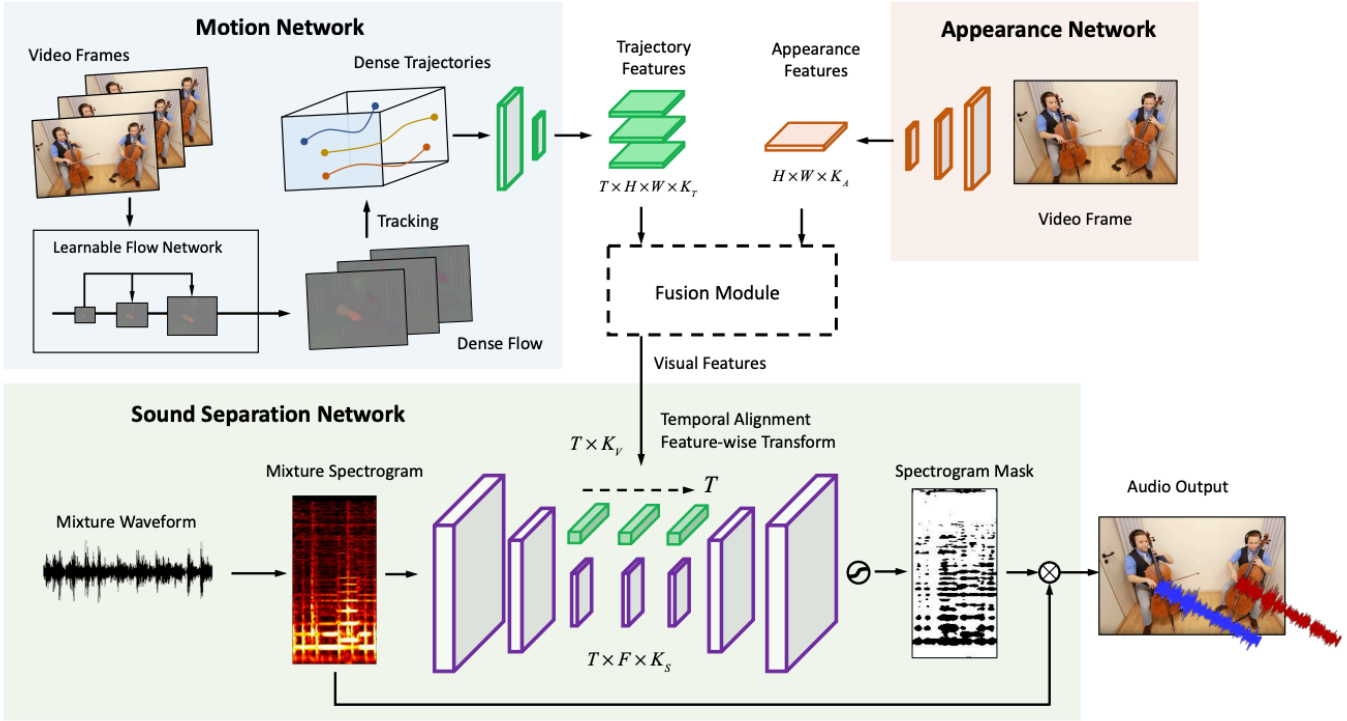


Source: Zhao et al. 2018

What If There Are Two Identical Instruments?

[Hang Zhao et al., The Sound Of Motions, ICCV (2019)]

Similar to Sound of Pixels, but adding the output of a motion network to the input of the sound separation network.

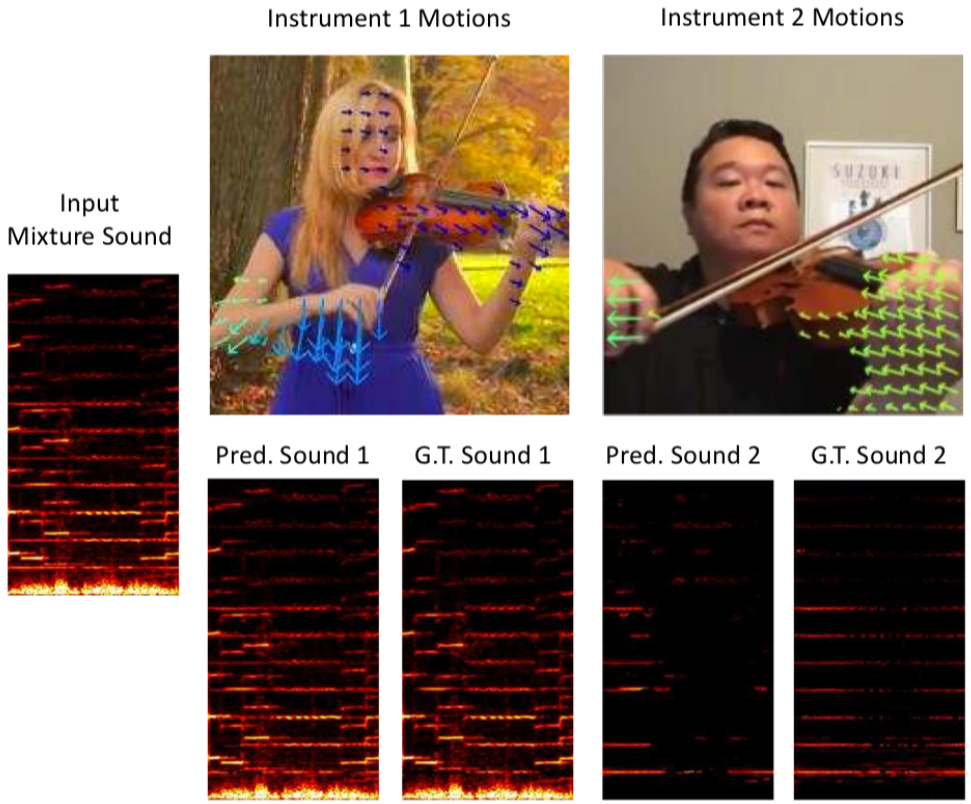


Source: Zhao et al. 2019

What If There Are Two Identical Instruments?

[Hang Zhao et al., The Sound Of Motions, ICCV (2019)]

Training via
mix-and-separate:



Source: Zhao et al. 2019

Patterns

- Remove words from your text to create a training signal
- Remove spatial or dimensional parts of your image to create a training signal
- Output a distribution instead of a value to avoid boring results
- Human supervision = selecting the right data to train on
- Use computed data or other sensors as training signal
- Use corresponding and non-corresponding examples for training
- Max-pool for training with a yes/no answer, use spatial information for inference
- Mix and separate

Summary

- In semi-supervised learning, only part of the data is labelled, or the labelling is of a different kind than would be needed for supervised learning.
- In self-supervised learning, the data provides the supervision. Usually, some sort of correspondence is used to force the network to learn good representations, e.g.
  - Spatial (image patches)
  - Multi-modal (image - sound)
  - Multi-sensor (RGB - depth)
  - Continuity of time (object tracking)
- Could eventually become much better than direct supervision (more data), but isn't yet there across the board.
- Currently, rapid progress is being made.
- Enables state of the art in NLP.

Questions?

Twitter: @chris_bonitz
Linkedin: linkedin.com/in/christoph-bonitz