I-Form

Advanced Manufacturing Research Centre

A World Leading SFI Research Centre

Science Foundation Ireland For what's next

Image (and Video) Analysis for Additive Manufacturing

Dr. Alessandra Mileo, Assistant Prof. *Xiao Liu, PhD student*

School of Computing, Dublin City University

HOST INSTITUTION

PARTNER INSTITUTIONS

FUNDED BY:

I-Form is an SFI Research Centre funded under the Science Foundation Ireland Research Centres Programme and co-funded under the European Regional Development Fund.

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Supervised, semi-supervised unsupervised

Supervised

(x,y) pairs → Given observation/input x, predict label/output $y = f(x)$

Semi-Supervised

Unsupervised

Set of observations x in X → Predict output y looking at the distribution of x

Supervised, semi-supervised unsupervised

Supervised

(x,y) pairs → Given observation/input x, predict label/output y = f(x)

Semi-Supervised

Set of observations x in X → Predict unknown output y given x

But, for a few observations *x' in X* you know the output/label *y*

Unsupervised

Set of observations x in X → Predict output y looking at the distribution of x

Deep Supervised Learning Examples

House pricing prediction (standard DNN)

Photo tagging (CNN)

Image Classification (CNN)

Machine translation (RNN)

Speech recognition (RNN)

…

In general where we have ***a lot of labelled data*** *(and resources) to train a deep learning model (from scatch)*

Deep Unsupervised Learning

Unsupervised approach

- Becoming very popular today… can you tell me why?
- Works under specific assumptions
  - smoothness
  - cluster
  - manifold

Application for unsupervised learning

- Clustering
- Feature selection
- Dimensionality reduction

Approaches to Deep Unsupervised Learning

Non-linear manifold (how do you approximate non-linear features distribution?)

- Autoencoder
- Restricted Boltzmann Machines
- Deep Belief Networks
- Generative Adversarial Networks

Deep Semi-supervised Learning

Myth

You cannot do deep learning without millions of labelled examples for your problem

In practice

- Learning from unlabelled data (**unsupervised** learning) is a flourishing research area
- **Pseudo-labeling** is another technique where you can learn from labelled and unlabelled data (semi-supervised)
- You can **transfer** representations learned from other related task (more on this later)
- You can use approaches such as **data augmentation** and **active learning** to have more (and better) data to train your model on (more later)

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Why do we look at Case Studies?

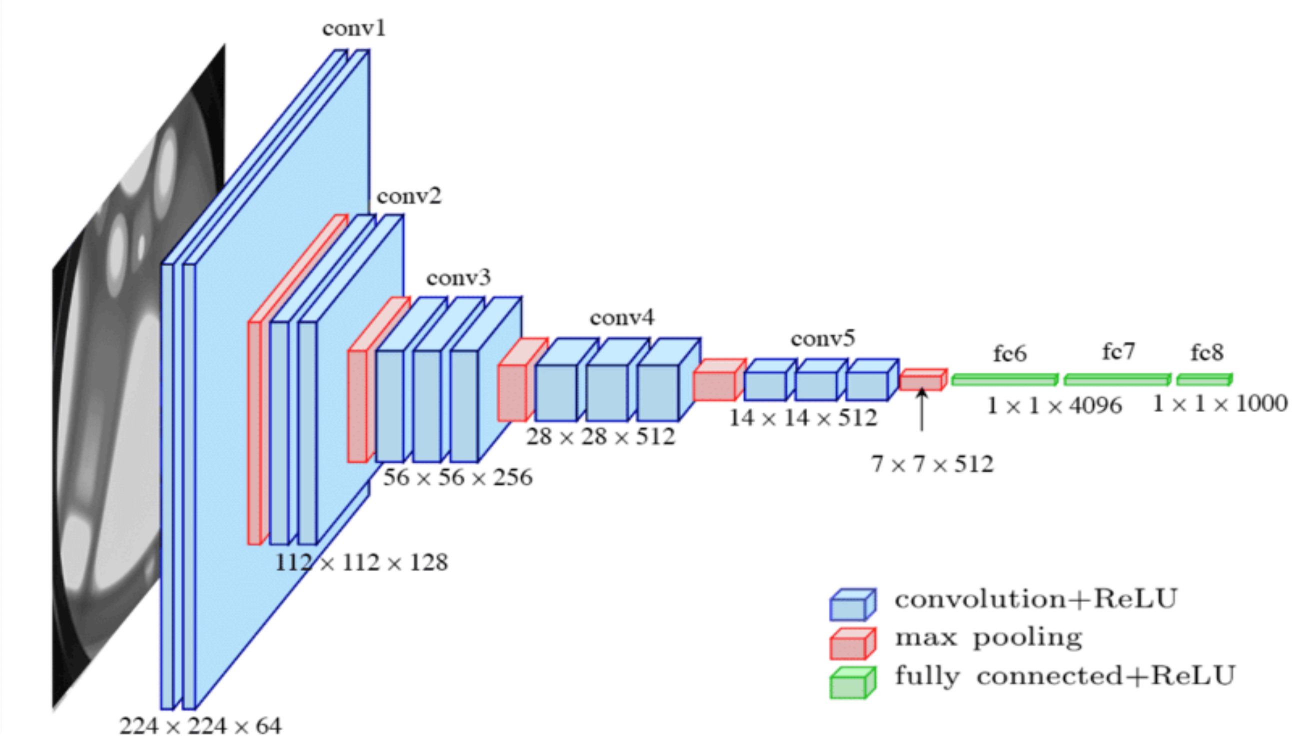
Plenty of research outputs/experiments on how to put together CNN building blocks to create effective ConvNets

- we can learn from the way they have done it
- turns out ConvNet architectures that are good for a given task, are also good for other related tasks
- most of the time, new applications of ConvNets happen by taking existing architectures and do minor changes to them (more later)

VGG(-16)

Novel Features compared to previous networks

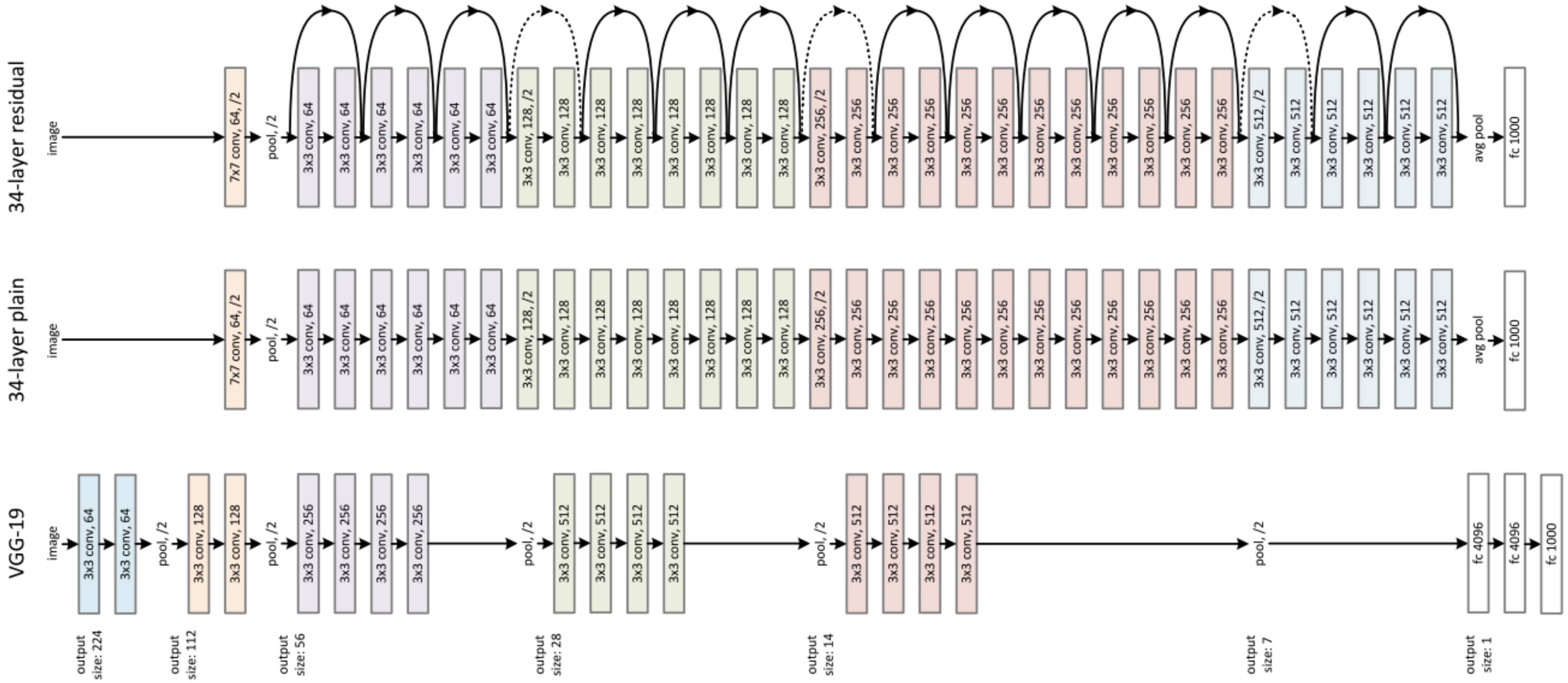
- Small convolutional filters (3x3 compared to previous, like Alexnet)
- Three 3x3 stacked together
    - three ReLu: more discriminative
    - less parameters
- Double the filters after each max-pooling layer (reduce dimensions but increased volume)



ResNet

Features

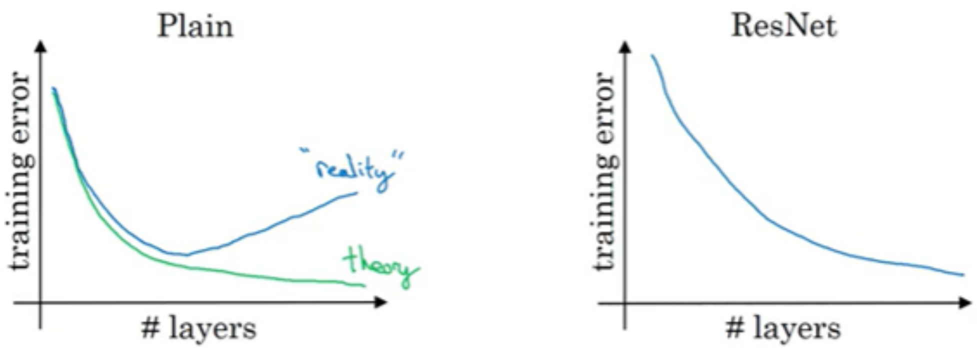
- Stacks residual blocks
- Residual blocks allow to skip some layers with a shortcut to inject actications at a given layer L to deeper layers directly
- Note use of same convolution to preserve dimension so I can do those skips
- For pooling layers(change dimension) multiply activation with a matrix of suitable dimensions (e.g. fixed matrix for zero padding, matrix of weights to be learned)



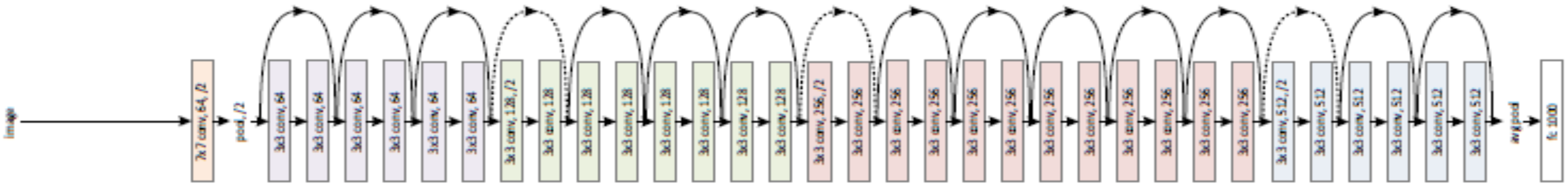
ResNet: why does it work?

Training error

With ResNet you can train more powerful, deeper network targeting the degradation problem that makes your error increase



AndrewNg



Why does it work?

- The extra layers are added as residual block (mapping to identity function)
- This does not add any extra parameters to be learn
- Let gradient descent improves accuracy through the extra backprop steps given by the added layers

Variants and Extension

- ResNet 18, 34 (residual blocks are 2 layers deep)
- ResNet 50, 101, 152 (residual blocks are 3 layers deep)
- DenseNet tackles the vanishing gradient by adding extra connections within each "dense block" (connecting each layer directly with each other in a block)

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

   In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

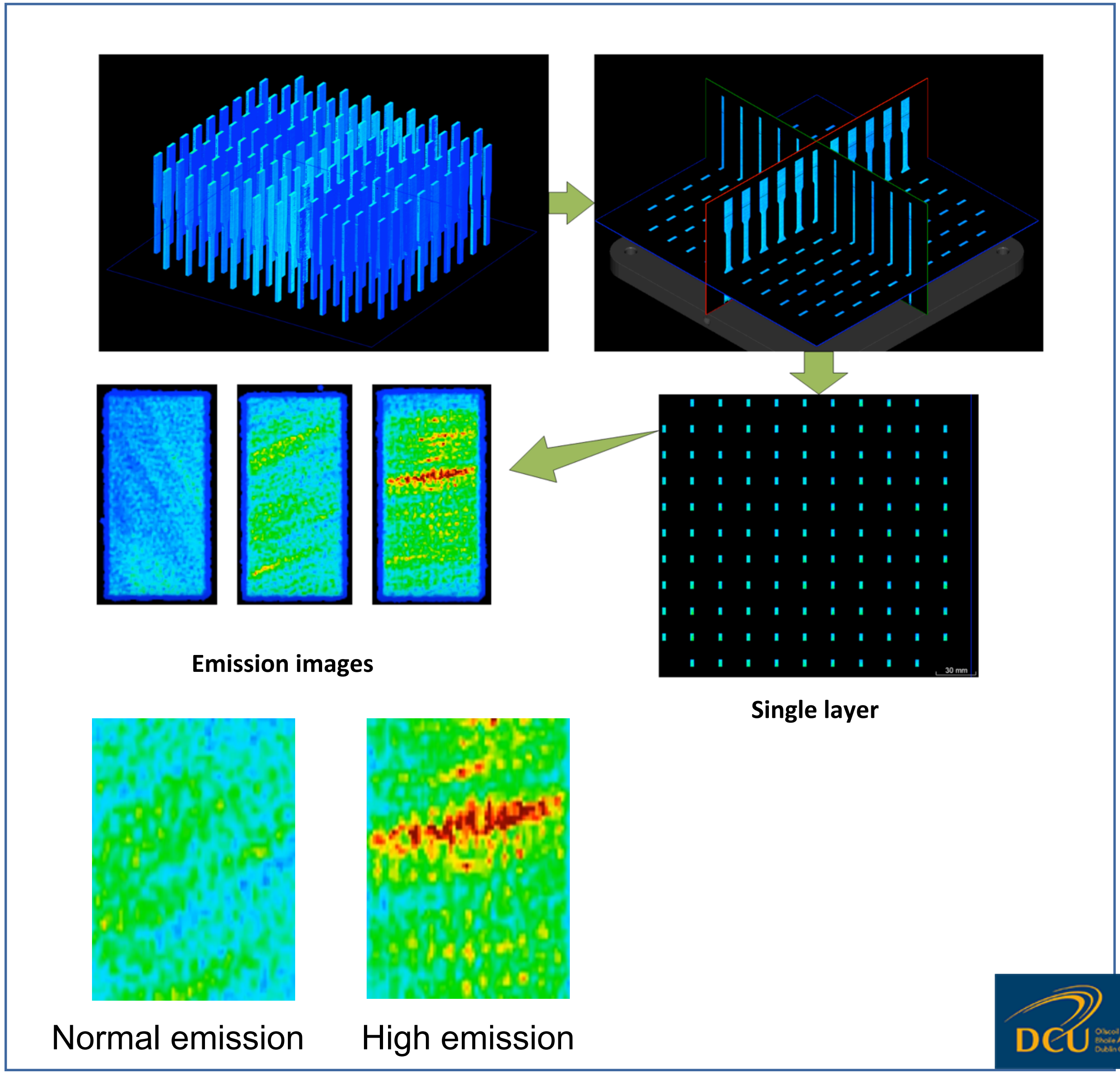
   In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Renishaw InfiniAM in-situ monitoring

A bit of context

- InfiniAM in-situ monitoring generates **A LOT of data** during the printing process
- This data includes information about meltpool emissions, in form of processed images (one for each layer)
- The formation of defects in parts is typically related to the stability of the meltpool: more instability (rapid heating/cooling in the process) is likely to generate defects
- It is still a manual process to do analysis and characterization of emissions, which involves looking at the 2D & 3D representations



Defect detection as an image classification problem

Note on data:

- No access to low level features, our dataset was only emision images!

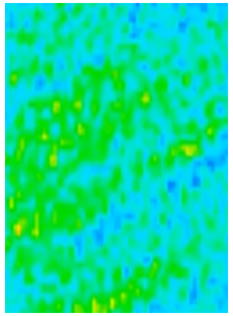
...

- No labelled data samples
- A lot more normal than high emission samples
- No information as to how the parameters were transformed into images
- ...

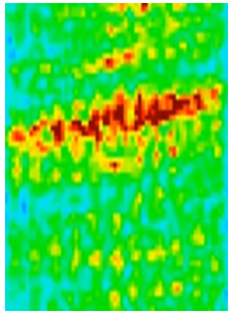
More on datasets later!

Our research questions:

1. Can we automatically classify all those (layer by layer) emission images to identify possible defect/no defect?
2. How early can we detect this?
3. Can we use our analysis to improve the quality of the data available for this task?
4. Can we assess the quality of the Renishaw monitoring system in generating those images (i.e. the correlation of low level features and images with properties of the build?)
5. How many different defect classes can we characterize?



Normal emission



High emission

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: ResNet and VGG-16
3. Defect detection in AM as an image classification problem

In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

   In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

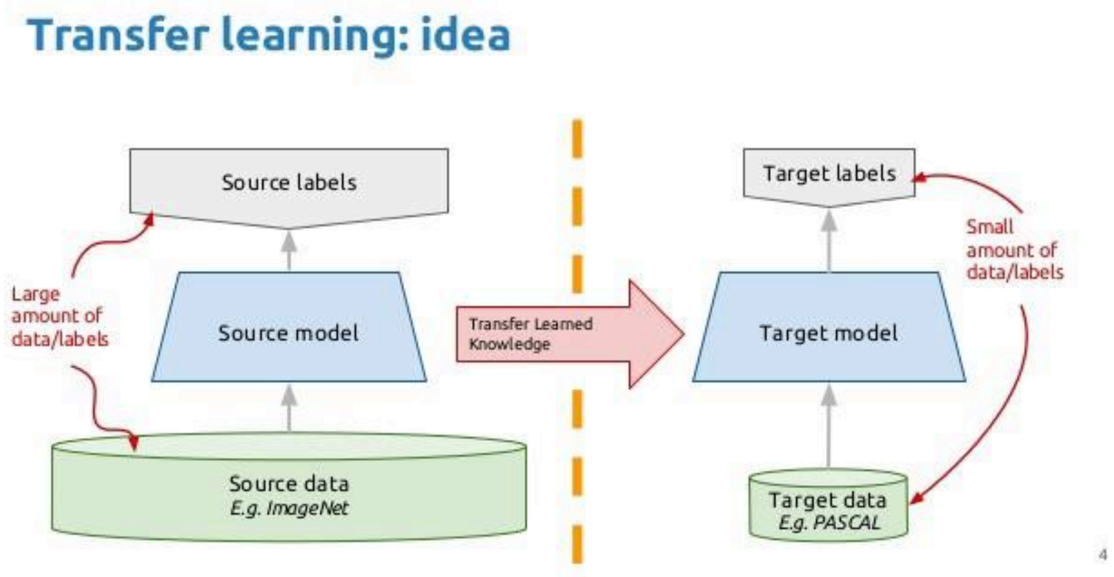
   In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Transfer learning

The general idea

- Take a network trained on a huge dataset (e.g. ImageNet, CoCo, CFAIR-100, MNIST, ...)
- This network has already learned weights for a particular **source task**
- You only need to adapt it to your domain and **target task** (for which you normally have a small set of labelled data)



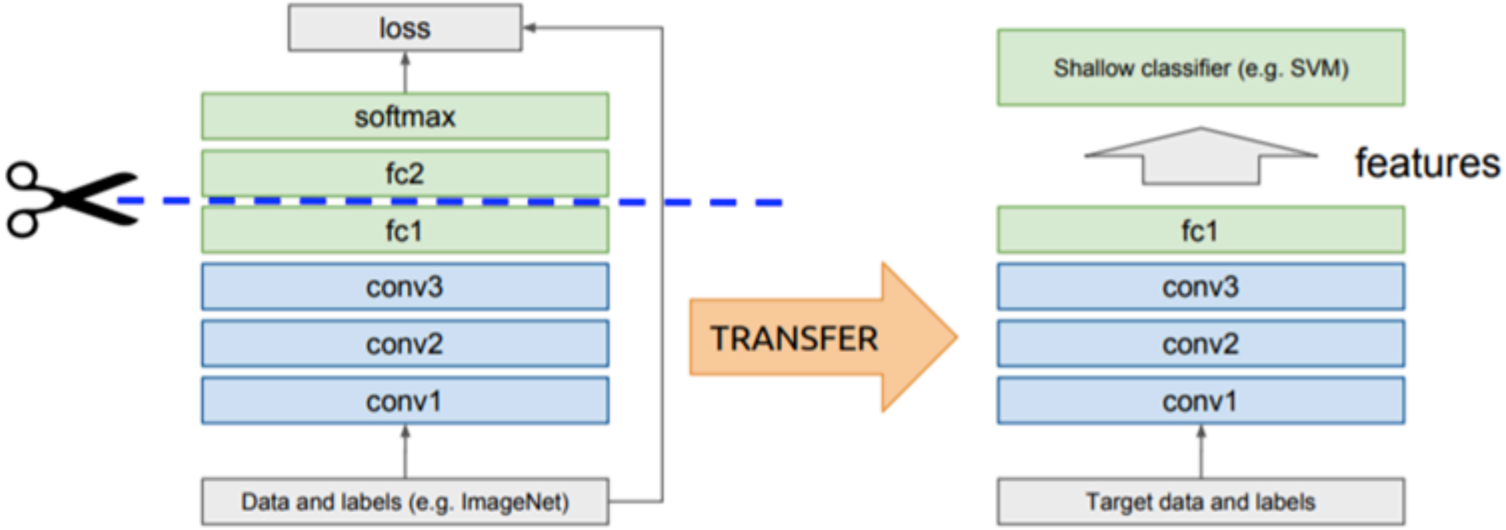
How do we do that?

Pre-trained models: https://keras.io/api/applications/

Transfer learning with off-the-shelf networks

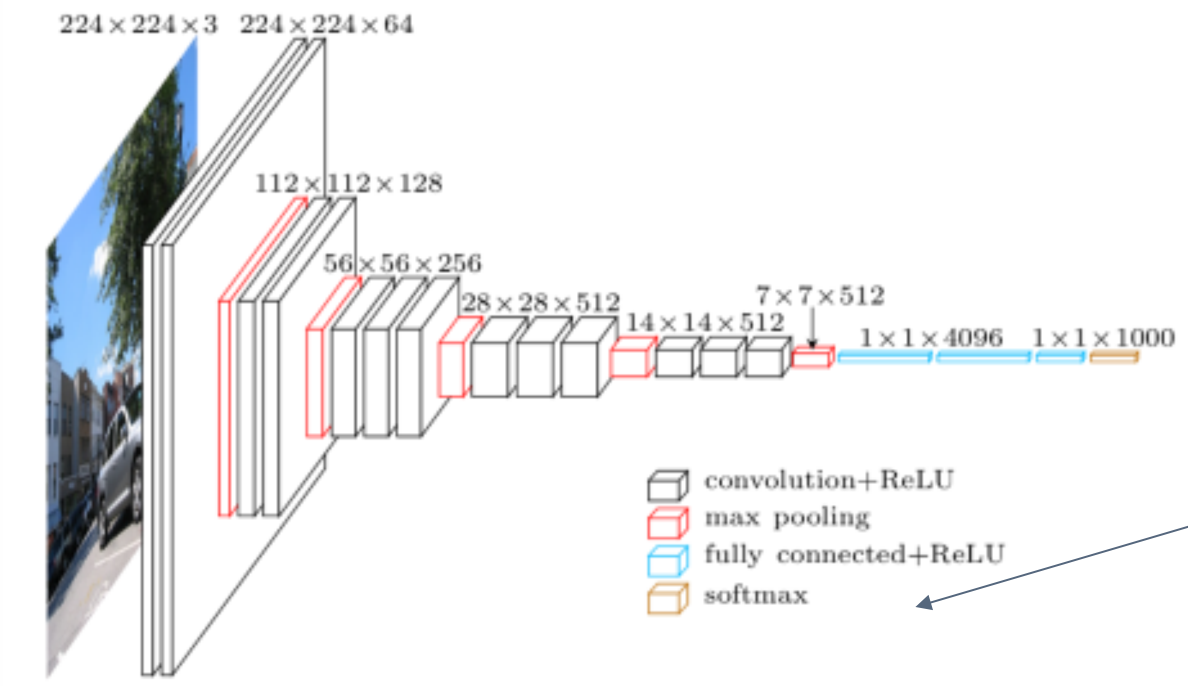
Idea:

- Take some (lower) layers of a network to detect generic features (e.g. edges and patterns) for a task (e.g. image classification)
- Train a shallow classifier (e.g. SVM) on those features
- Note: **lower layers** transfer well to other tasks as they are low level features, while **higher layers** are more task specific



Razavian et al. CNN Features off-the-shelf: an Astounding Baseline for Recognition, CVPRW 2014. https://arxiv.org/abs/1403.6382

We can do better: supervised task adaptation

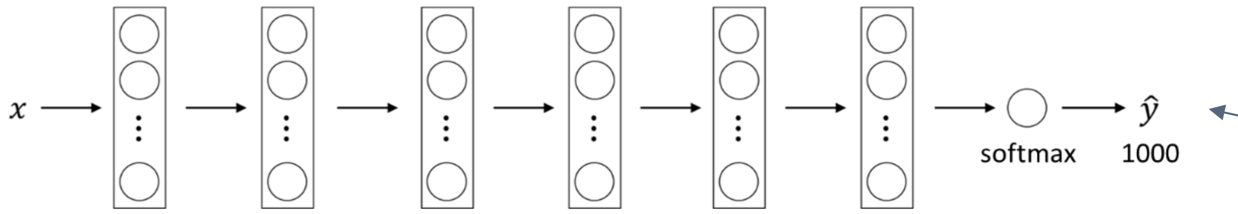


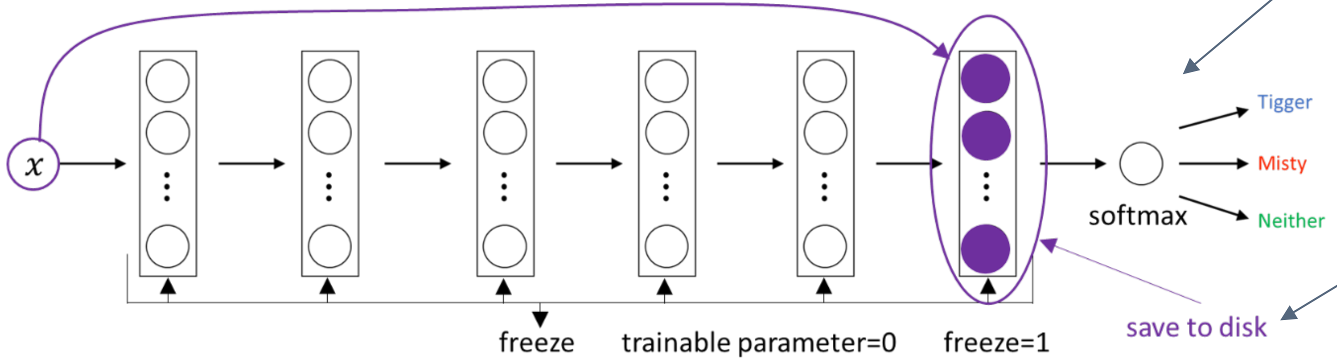
Softmax

Gives a probability distribution of the label candidates or list of classes. Usually it is the last layer in a classification task.

Softmax is strictly related to the problem you want to solve

We can do better: supervised task adaptation

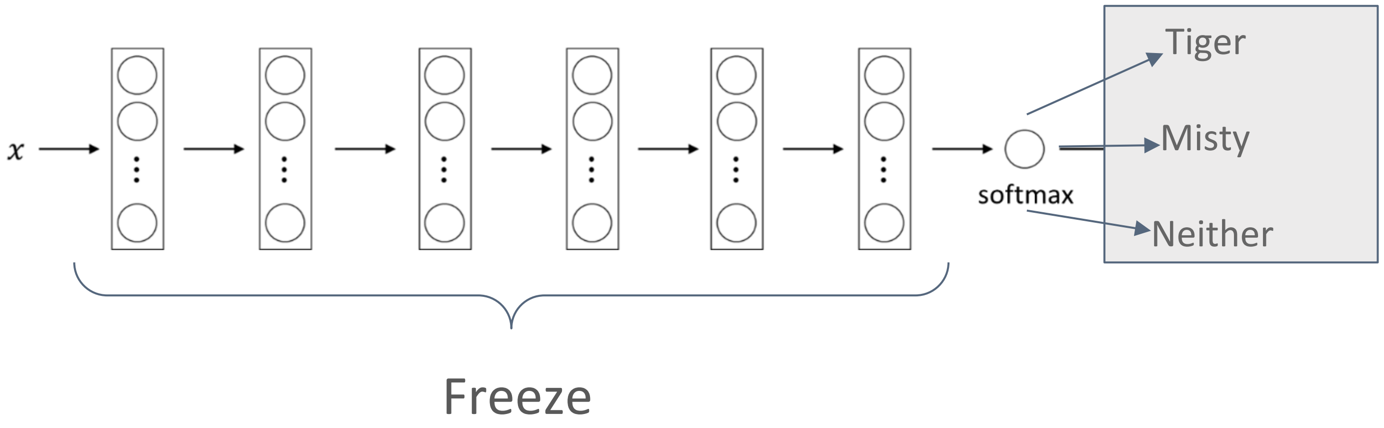


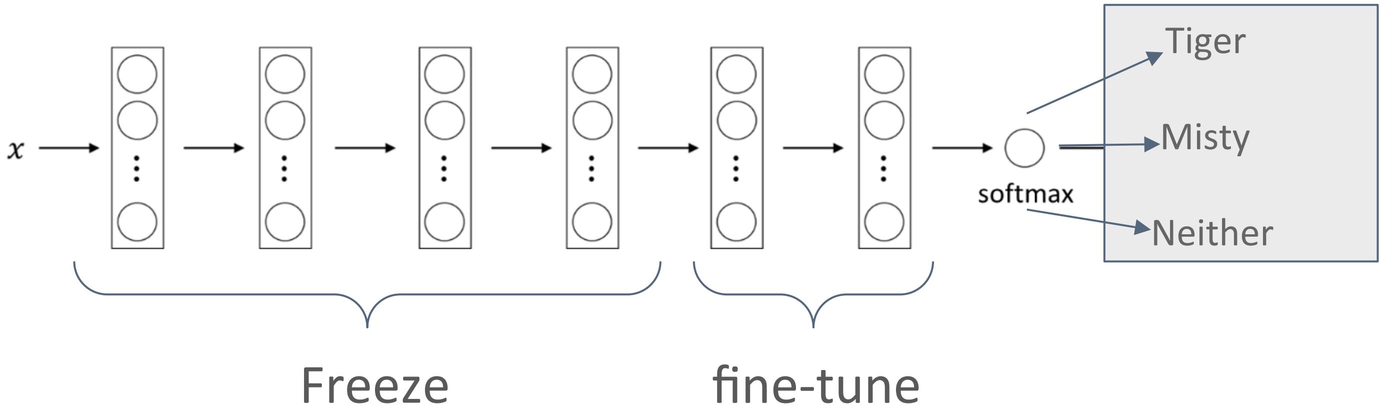


General approach

- Network trained to classify images in 1000 classes (ImageNet)
- I want to use this network to classify images of my cat (**Tigger**) and my neighbours cat (**Misty**)
- First thing I do is replace that softmax layer to classify among my 3 classes
- We freeze the hidden layers of the **source trained network** (e.g. those weights do not get learned/updated during backpropagation)
- For faster implementation I can save the activations at the last layer to disk.

We can do even better (sometimes): Fine tuning





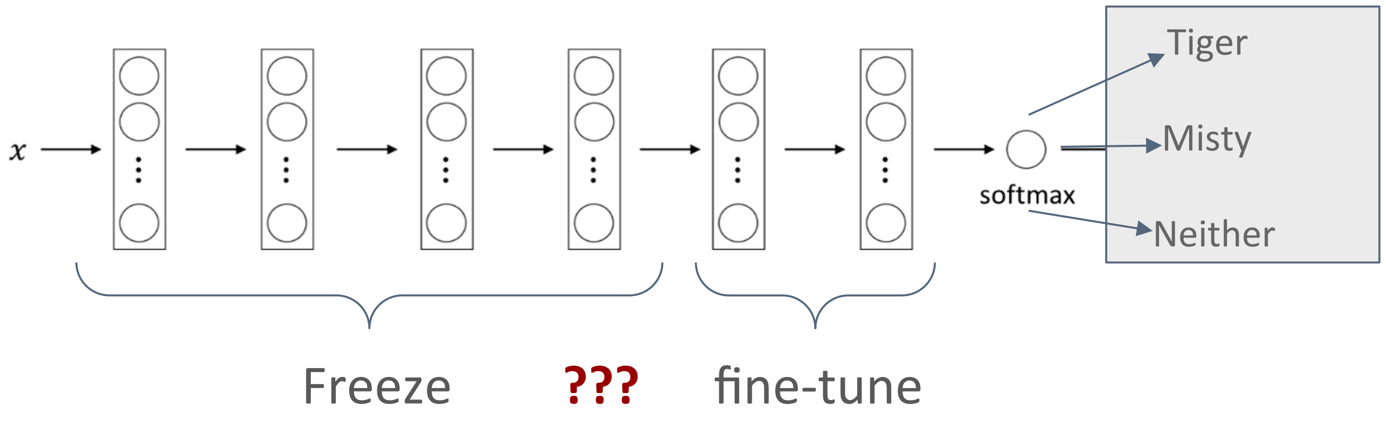
The general idea

- I don't need to freeze all layers
- Some high level layers can be trained using backpropagation on the target labels (fine-tuned)

Fine-tuning options

- fine-tune 1 or more layers
- fine tune starting from random weights
- fine-tune starting from trained weights

Freeze or fine-tune?



It depends on target task:

- Are target labels scarce? Freeze as many layers as you can (possibly all)
- Do I have more target labels? Freeze less layers
- it is a trade-off, and it is normally found empirically (or looking at what others have done)

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

   In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

   In practice: Transfer learning and fine tuning in Keras

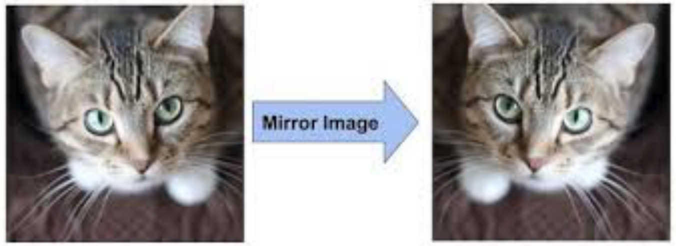
7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Enlarge your dataset with data augmentation

When can it help in general?

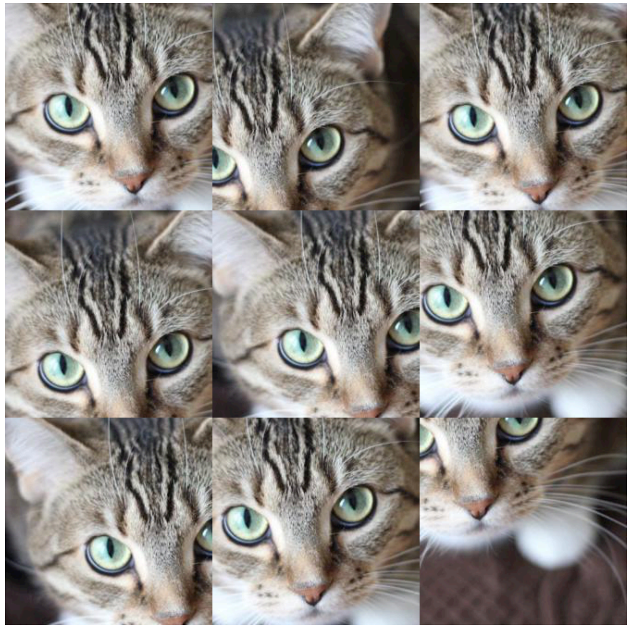
- Train Deep Networks from scratch
- Use transfer learning and fine tuning
- Deep semi-supervised or supervised learning in general

Mirroring

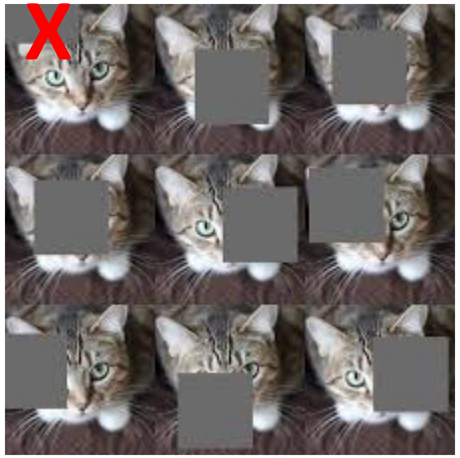


- Horizontal/vertical mirroring preserves what you are trying to recognise
- In general if the image on the left is a cat, the mirrored one is still a cat

Random cropping



What possible problem do you see with this technique?

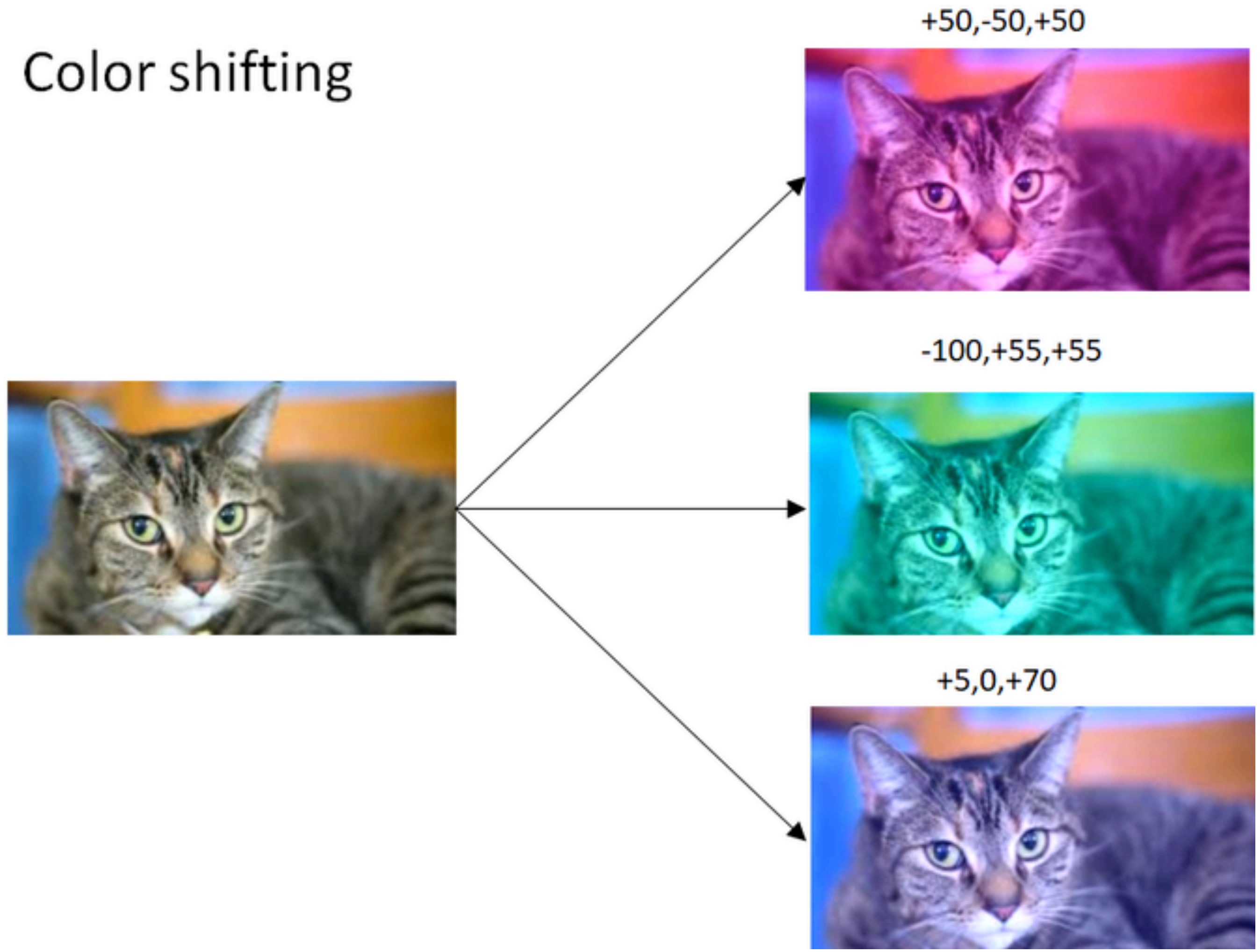


Need a sufficiently large portion of the image in the crop

Less common techniques

- Rotation
- Shearing
- Translation
- Local Warping
- ...

Enlarge your dataset with data augmentation



Color shifting

- Add some distortorsion to any of the R G B channels
- In practice color can change (e.g. in sunlight or shades or other conditions) but the label doesn't

Note on Implementing color shifting

Often done using PCA colour augmentation to determine what RGB channel to add more distortion to

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

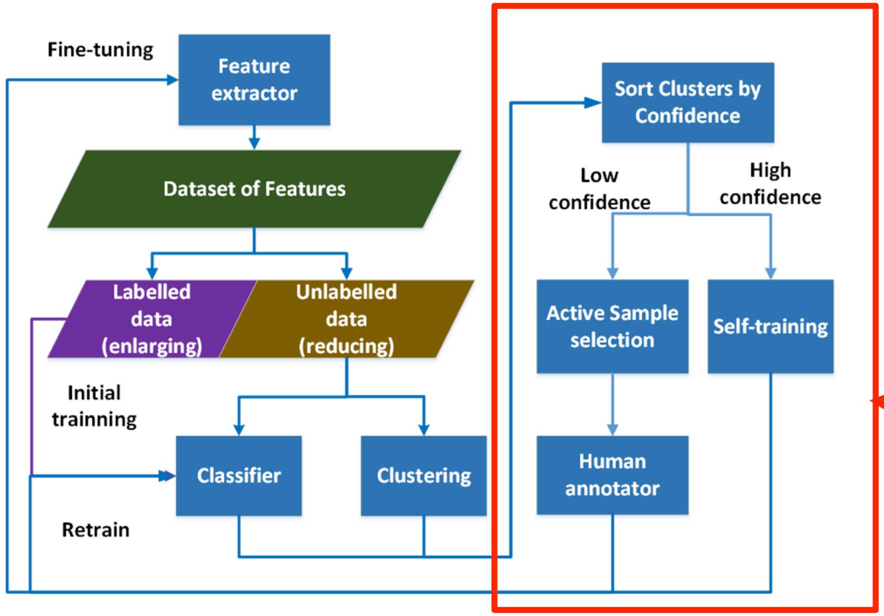
In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Next steps and open challenges



Active Learning (work in progress)

- Interactive approach
- Select examples to label for improved accuracy (e.g. low confidence samples)
- Based on several heuristics (e.g. representativeness and informativeness of the sample)
- Reduce labeling cost, improved accuracy

Our research questions:

1. Can we automatically classify all those (layer by layer) emission images to identify possible defect/no defect?
2. How early can we detect this?
3. Can we use our analysis to improve the quality of the data available training our model?
4. Can we assess the quality of the Renishaw monitoring system in generating those images (i.e. the correlation of low level features and images with properties of the build?)
5. How many different defect classes can we characterize?

Ongoing experimental evaluation and comparison (work in progress)

- Comparing our classification results with results obtained from low level data
  - CT-scan
  - time series analysis
- Objective is to verify convergence and also test the sensitivity of InfiniAM monitoring in generating emission images

Additional research question (more in the next slide)

Would the approach be better at detecting different types of defects as early s possible if we added the temporal aspect? (i.e. video analysis)

Outline

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

   In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

   In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Images vs. Videos

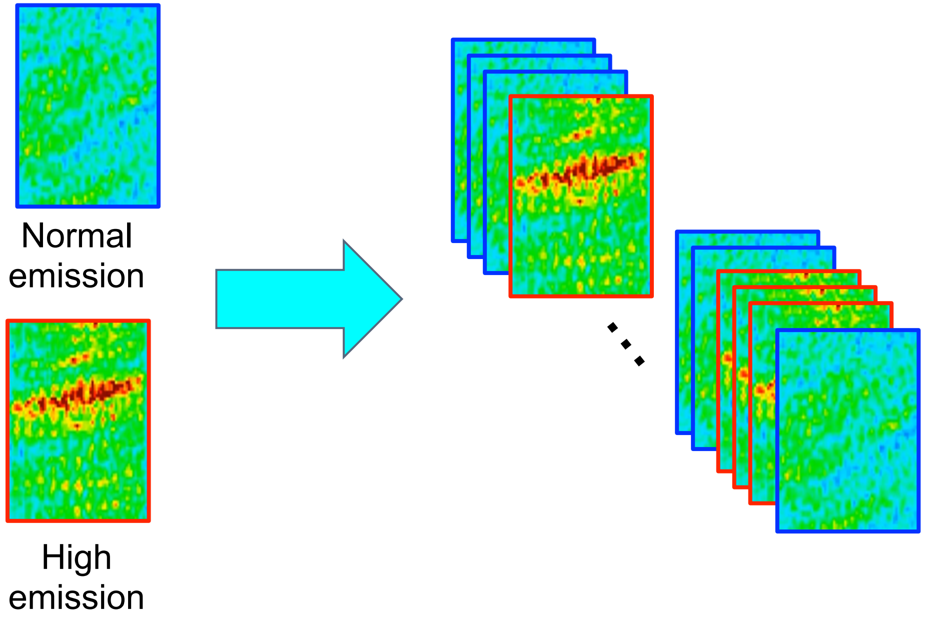
What is a Video?

- Video as a 3D signal: coordinates x, y and temporal dimension t
- if we fix t, then we have an image
- for a discrete sequence of t, a video is a sequence of images (frames)

Question

How to extend CNN for images analysis to images sequences (= videos) ?

Why are we interested in this?



We have not looked into it (yet)

Deep Video Analysis: Basic Deep Architectures

Single frame model	Each frame is processed by a separate CNN (**in parallel!**), they are then combined with some pooling (but pooling is not aware of the temporal order, which is therefore **lost**!)
CNN + RNN	Recurrent Neural Networks, unlike feed forward networks (like CNN) can learn sequences as the output depends on previous states (note RNN are sequential, cannot be parallelised)
3D Convolutions	Add an extra dimention to standard CNN (which represents time), so that your convolutional filters are 3D. The video is split into segments, where the number of frames in each segment corresponds to the temporal dimension of your 3D conv. filter
Two-stream model	Combines a spatial stream and a temporal stream ConvNet, trained independently (with separate softmax). The two outputs can be averaged or concatenated and passed to an SVM classifier

Various combinations have been proposed. This is an active area of research!

This conclude our journey

1. Supervised, semi-supervised and unsupervised (deep) learning
2. CNN case studies: VGG and ResNet
3. Defect detection in AM as an image classification problem

   In practice: Dataset exploration in codelab (*InfiniAM and DAGM*)

4. Transfer learning and fine tuning
5. Data augmentation
6. Defect detection in AM: our approach

   In practice: Transfer learning and fine tuning in Keras

7. Next steps and challenges for Image Analysis in AM
8. Deep Learning for Video Analysis: an intuition

Q & A

Resources

- Pre-trained Deep Neural Networks:
  https://keras.io/api/applications/
- Keras tutorial on transfer learning and fine tuning:
  https://keras.io/guides/transfer_learning/
- Book on Advances in Deep Learning, Springer
- An overview of Semi-Supervised Learning:
  https://arxiv.org/pdf/2006.05278.pdf
- CNN vs SIFT papers: Yan et al., 2016, Zheng et al., 2015

Note: for a good introduction to Deep Learning and CNN you can also use resources like Coursera (esp. courses by Andrew Ng)