TCL Research Europe

MIMUW cooperation

About TCL

Television Brands

TCL FFALCON THOMSON TOSHIBA ROWA Melodi

Mobile Phone Brands

TCL alcatel BlackBerry palm

- Established in 1981
- >75,000 employees worldwide
- $17bn annual income
- 3rd largest TV producer
- 27 research centers including 5 AI centers (3 in China, Hong Kong and Poland)

TCL Research Europe

New research center with the objective to research new Artificial Intelligence technologies and apply results to global products of TCL Group

- Office in Warsaw, Poland
- Officially since September 2018
- Planned employment of over 100 AI researchers and engineers (15 now)
- Collaboration with regional universities

Computer Vision

NLP

On-Device AI

We offer

- Non-corporate / startup culture
- Mentoring by experts in AI (NLP, computer vision, machine learning, data science, ...)
- Computing GPU infrastructure
- Real-world AI challenges
- Opportunity to apply research results to global products
- Job offer or PhD grant
- Possibility of international exchanges (e.g.: Honk-Kong, China)

Projects

Projects – Network optimizations

Optimizing network running time on device

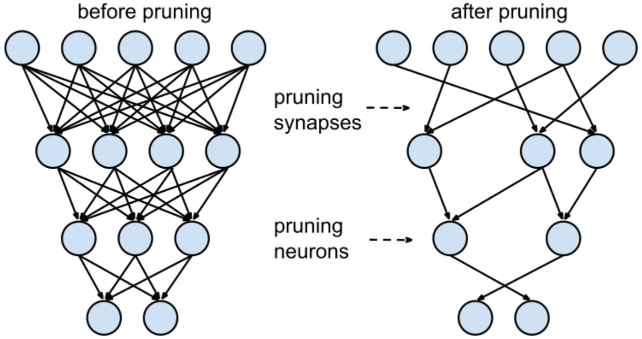
$$FPS \propto SOFTWARE\ EFFICIENCY * ALGORITHM\ EFFICIENCY * HARDWARE\ EFFICIENCY$$

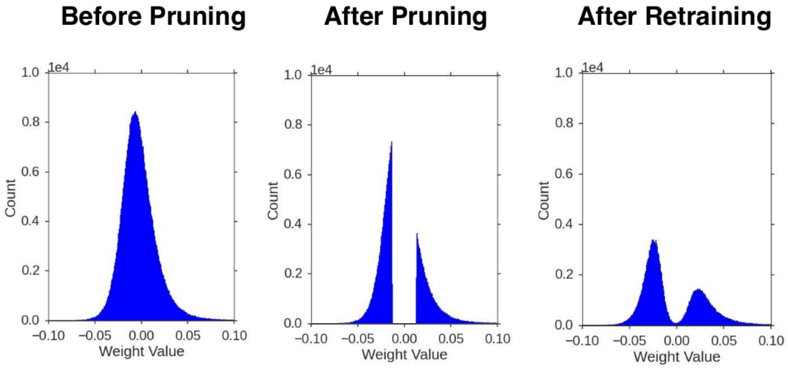
$$FPS \propto LIBRARY\ EFFICIENCY\ COEFFICIENT * \left(\frac{DEVICE\ GFLOPS}{MODEL\ GFLOPS}\right)$$

1) Use / improve / create specialized libraries for running neural networks on embedded devices
2) Improve models so that they are smaller and require less operations (e.g. pruning and retraining, fp16 instead of fp32)
3) Use hardware specialized towards neural networks (i.e. having certain matrix operations implemented efficiently)

Improve models – more efficient algorithms *

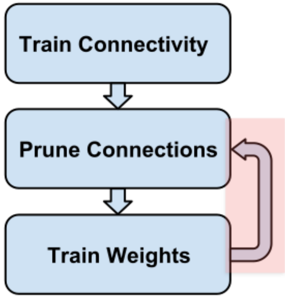
Pruning models (especially FC layers) – removing unimportant weights (centered around 0)

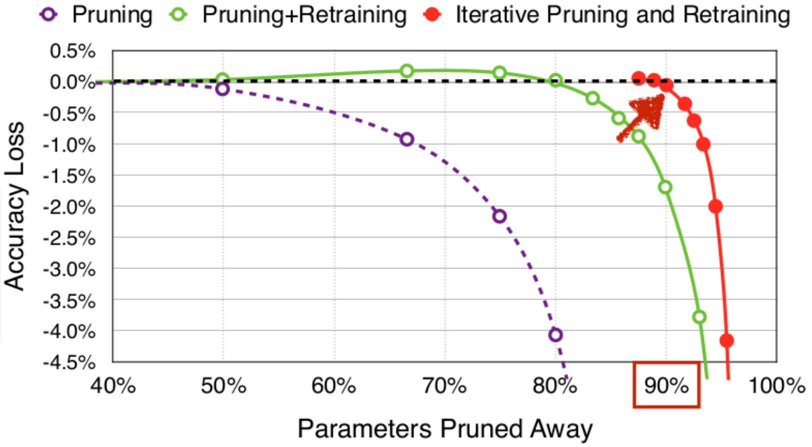




[Lecun et al. NIPS'89]

[Han et al. NIPS'15]





* based on Song Han slides

Improve models – more efficient algorithms *

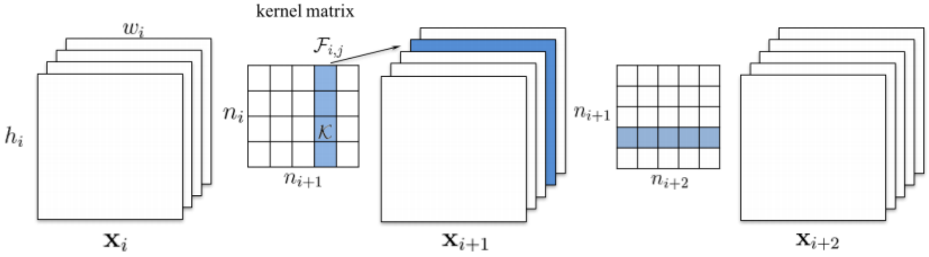
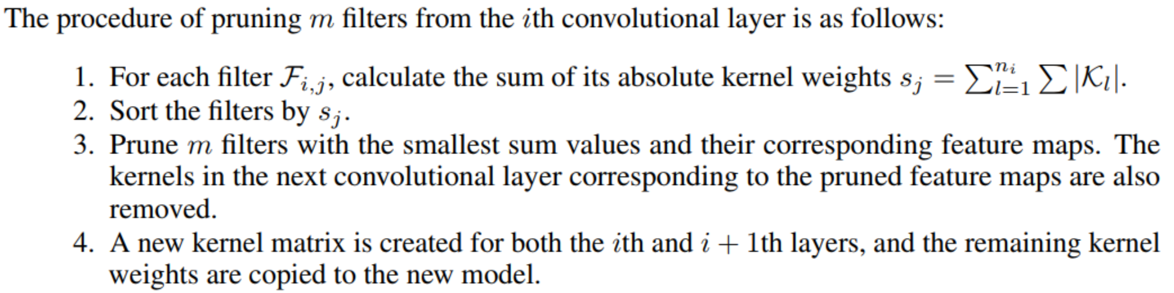
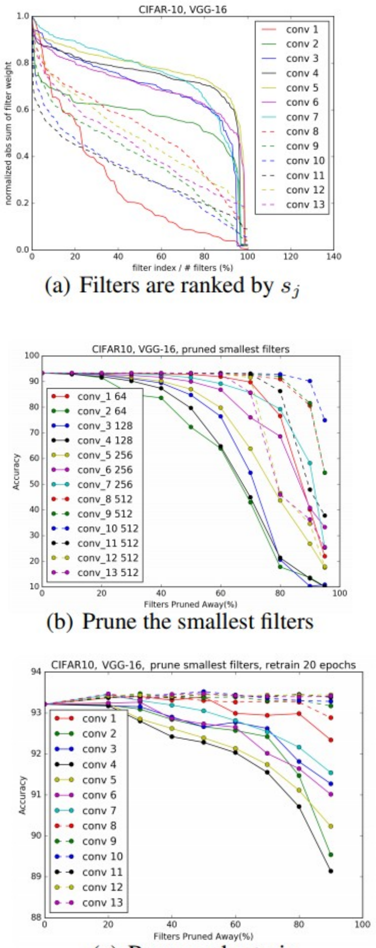


Figure 1: Pruning a filter results in removal of its corresponding feature map and related kernels in the next layer.



The procedure of pruning m filters from the ith convolutional layer is as follows:

1. For each filter $\mathcal{F}_{i,j}$, calculate the sum of its absolute kernel weights $s_j = \sum_{l=1}^{n_i} \sum |\mathcal{K}_l|$.
2. Sort the filters by s_j.
3. Prune m filters with the smallest sum values and their corresponding feature maps. The kernels in the next convolutional layer corresponding to the pruned feature maps are also removed.
4. A new kernel matrix is created for both the ith and $i + 1$th layers, and the remaining kernel weights are copied to the new model.



(a) Filters are ranked by s_j

(b) Prune the smallest filters

(c) Prune and retrain

* based on PRUNING FILTERS FOR EFFICIENT CONVNETS

Questions

- What is the relation between pruning ratio and accuracy change and running time
  - For which model types we can compress most without significant accuracy loss (image, text – convolutional, recurrent, transformer, etc.)
  - Which layers can be compressed most (convolutional, fc – bottom, top – are results consistent among similar models)
  - Is decrease in the running time proportional to the compression ratio? If not why (e.g. in case of TensorFlow lack of implementation of sparse operations which makes sparse fc layers not giving any speedup)
- TensorFlow prunning and retraining
  - Typical situation – somebody during training doesn't care about the efficiency – just about accuracy
  - Can we build a prototype which takes a trained model and its training data, converts the model (possibly using TensorFlow features) and retrains to compensate removed weights
- Efficient running sparse models
  - Can we integrate running sparse models (i.e. implement sparse multiplication or apply some ready linear algebra packages) into some of the open source libraries intended to run models faster (e.g. TFLite, MACE, NCNN, Tengine) and what speed-ups we can get
  - ➢ What speedup in models running time we can get there

Projects - NLP

Machine Reading Comprehension

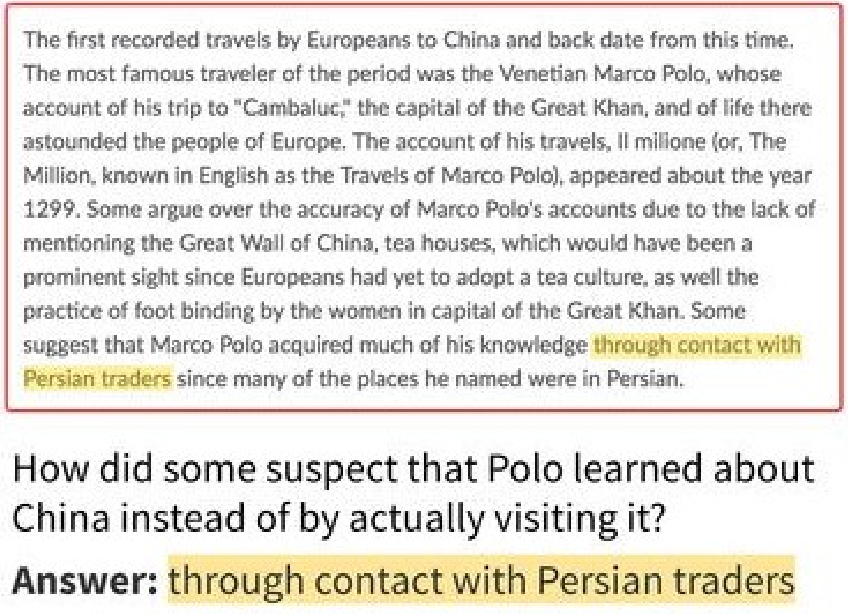
Goal: To answer questions based on piece of text.

Use cases

- Q&A systems, personal assistants, dialogue agents
- Text summarization, reports analysis

Competitions

MS Marco, Stanford SquAD, Facebook bAbl, TriviaQA



The first recorded travels by Europeans to China and back date from this time. The most famous traveler of the period was the Venetian Marco Polo, whose account of his trip to "Cambaluc," the capital of the Great Khan, and of life there astounded the people of Europe. The account of his travels, Il milione (or, The Million, known in English as the Travels of Marco Polo), appeared about the year 1299. Some argue over the accuracy of Marco Polo's accounts due to the lack of mentioning the Great Wall of China, tea houses, which would have been a prominent sight since Europeans had yet to adopt a tea culture, as well the practice of foot binding by the women in capital of the Great Khan. Some suggest that Marco Polo acquired much of his knowledge through contact with Persian traders since many of the places he named were in Persian.

How did some suspect that Polo learned about China instead of by actually visiting it?

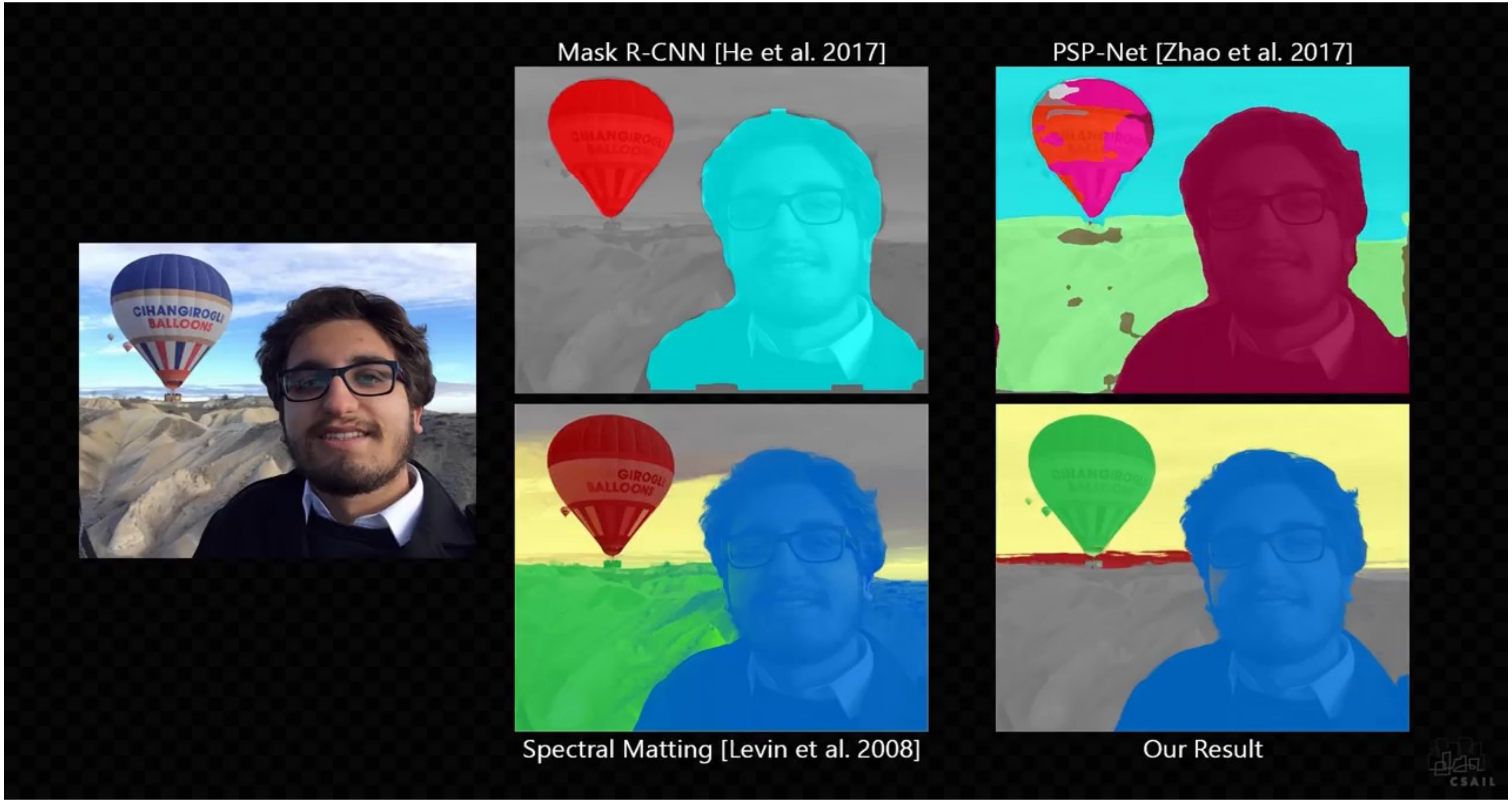
Answer: through contact with Persian traders

Source: https://rajpurkar.github.io/mlx/qa-and-squad/

Projects – Computer Vision

Semantic Soft Segmentation

(https://www.youtube.com/watch?v=QYIQbfnS9jA)

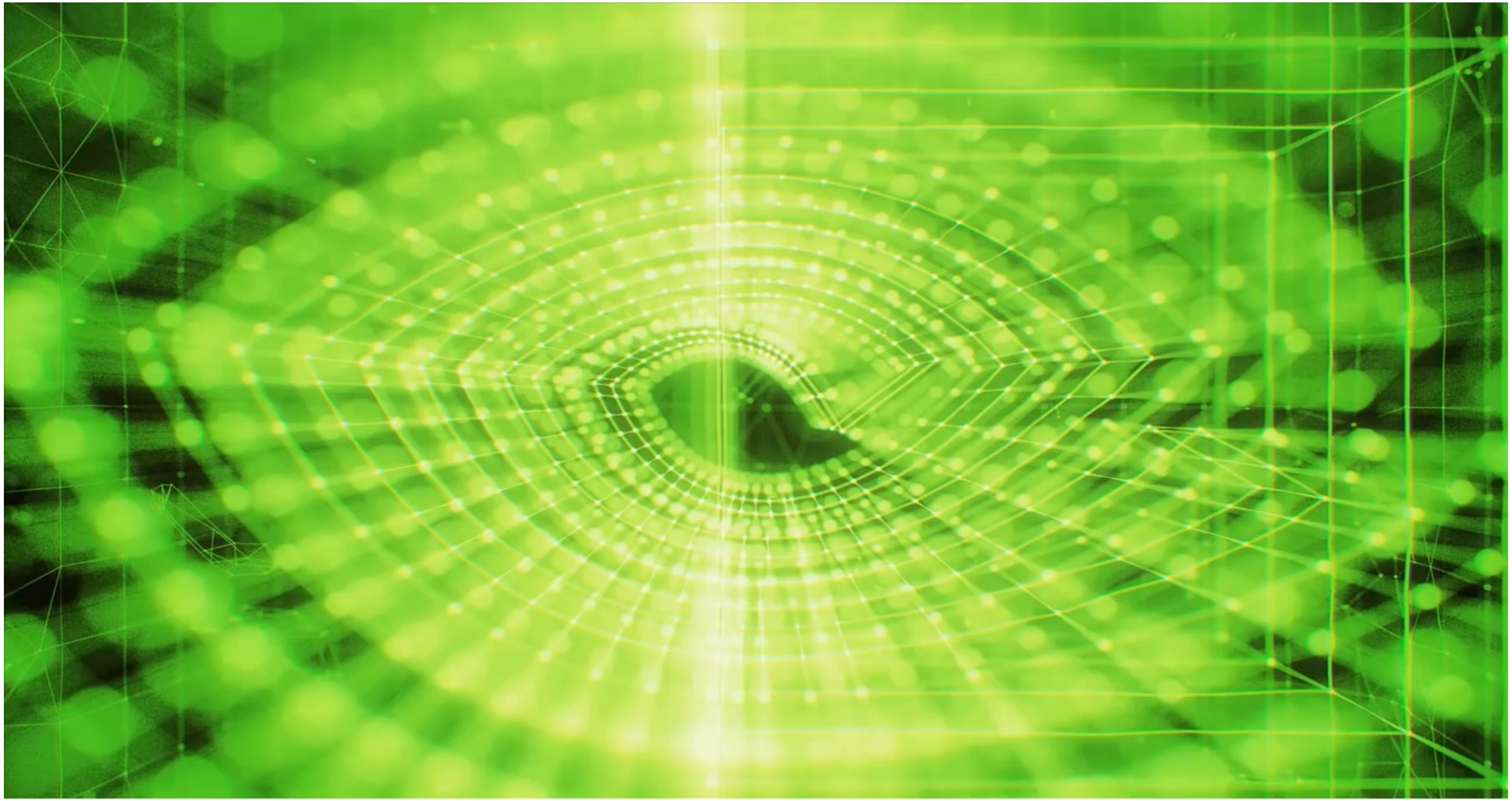

Mask R-CNN [He et al. 2017]
PSP-Net [Zhao et al. 2017]
CIHANGIROGLU BALLOONS
Spectral Matting [Levin et al. 2008]
Our Result
CSAIL

Semantic Soft Segmentation

- **Problem definition** – precise semantic segmentation, with accurate transitions between objects, applied for manipulating background and foreground objects
- **Possible application** – special effects in camera/gallery application, pointing objects for removal, etc.
- **Example** – “Semantic Soft Segmentation” paper
  - ➢ https://www.youtube.com/watch?v=QYIQbfnS9jA
- **Expected output:**
  - ➢ Replicating results from the paper
  - ➢ Proposing (and implementing) performance optimizations
  - ➢ Implementing a simple DEMO (e.g., background removal)

Video frame interpolation using deep neural networks

(https://www.youtube.com/watch?v=MjViy6kyiqs)

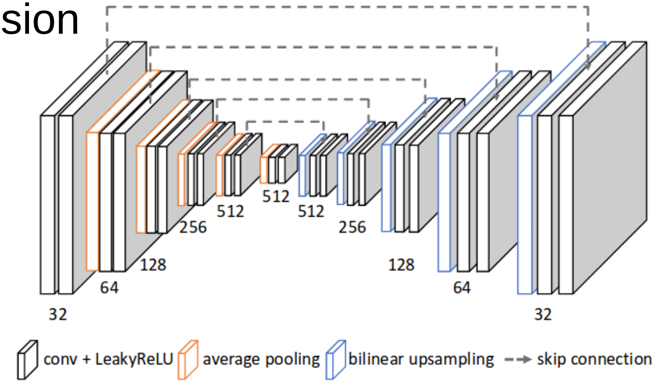


Video frame interpolation using deep neural networks

- **Problem definition** - generate intermediate frames between given frames in order to form both spatially and temporally coherent video sequences
- **Possible application** - slow-motion effects, frame rate conversion
- **Latest approaches** -mainly based on CNN networks
- **Example** from the latest NVIDIA paper

  Transforming Standard Video Into Slow Motion with AI:

  ➢ https://youtu.be/MjViy6kyiqs



Network structure from: *Transforming Standard Video Into Slow Motion with AI*

- **Expected output:**

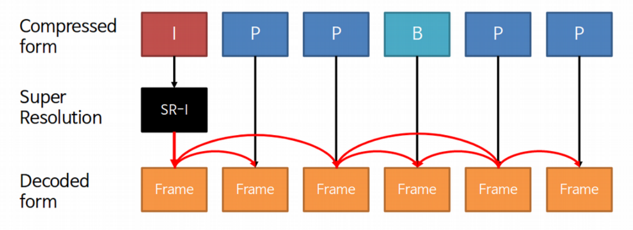
  ➢ Evaluation and comparison of a few selected approaches

  ➢ Implementation of an end-to-end pipeline based on the selected approach (preferably a demo application for an Android phone)

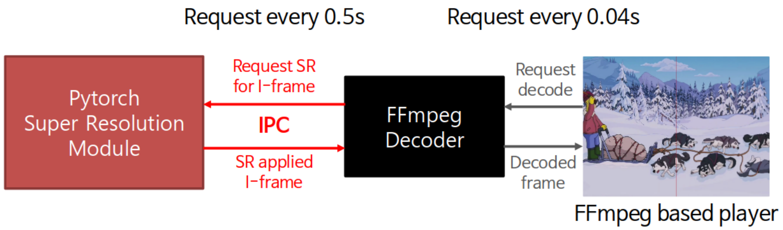
  ➢ Test results for a selected data set (e.g. ML slow-motion)

Codec based video enhancement using deep neural networks

- **Problem definition** – enhancement of the video stream, e.g. super resolution (reconstruction of a high-resolution (HR) video from its down-scaled low-resolution (LR) version, full HD to 4k/8k)
- **Possible application** – real time video enhancement
- **Latest approaches** – mainly based on deep learning
- **Example** – *RTSR: Enhancing Real-time H.264 Video Streaming using Deep Learning based Video Super Resolution*:
  - ➢ https://insujang.github.io/assets/pdf/cs570_final_ppt.pdf



- **Expected output:**
  - ➢ Evaluation and comparison of existing approaches
  - ➢ A custom decoder that runs the enhancement algorithm (e.g. key frames only + decoding of the other frames based on enhanced key frame)
  - ➢ Implementation of the proposed decoder as a library



from https://insujang.github.io/assets/pdf/cs570_final_ppt.pdf

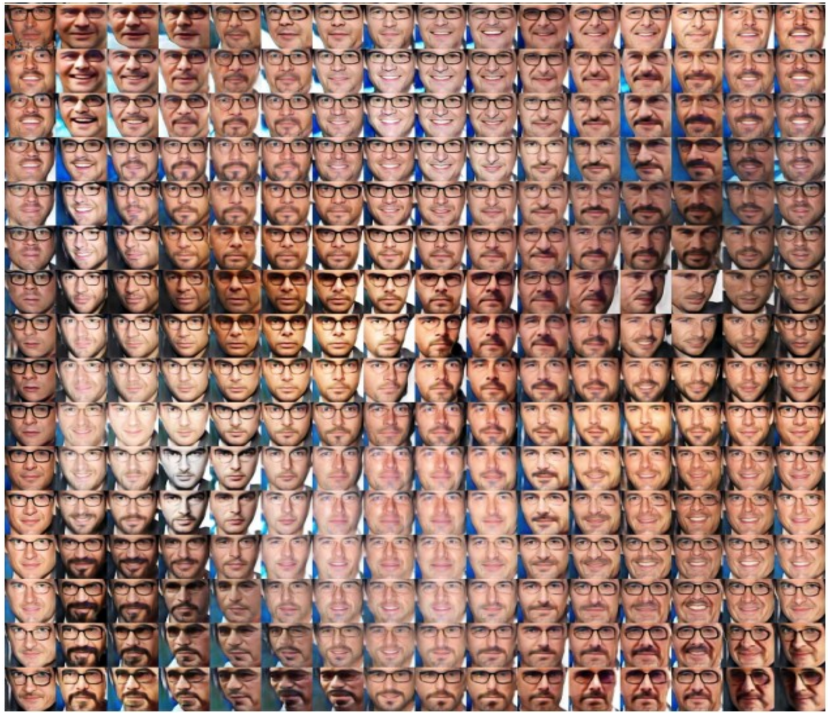
Improving video compression based on AI models

- **Problem definition** – apply AI models in video compression pipeline, e.g. for compressing key frames, motion modeling,
- **Possible application** – improving existing codecs (H.265, ...), designing new end-to-end codecs
- **Examples:**
  - *End to end video compression Video Compression through Image interpolation (*ECCV2018), https://chaoyuaw.github.io/vcii/
  - *Performance Comparison of Convolutional Autoencoders, generative adversarial Networks and super-resolution for Image Compression* (link)

- **Expected output: To Be Defined...**

Generative models for data augmentation

- **Problem definition** – using generative models to augment training data
- **Possible application** – one-shot learning, improving accuracy of any classifier, etc.
- **Latest approaches** – GANs, generators embedded into training pipeline
- **Examples:**
  - https://arxiv.org/abs/1711.04340 - conditional GANs
  - https://arxiv.org/abs/1801.05401 - generative models embedded into the training pipeline
  - https://arxiv.org/abs/1803.01229 - GANs for medical images
- **Expected output:**
  - test results for selected datasets and for selected problems
  - this may require implementing some modeling pipelines (see papers), e.g. in Tensorflow
  - summary: which generative models can be used and in what way for a certain types of problems?

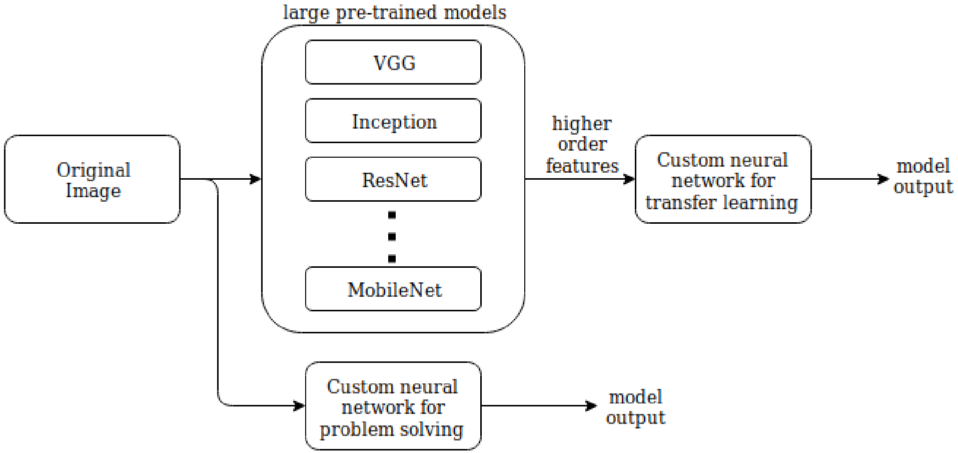


Transfer Learning In Computer Vision: Models Comparison

- **The main aim of the project:**

  Comparison of several transfer learning models (VGG, Inception, ResNet, MobileNet, etc) and their properities in different computer vision tasks: image classification, image generation, etc.

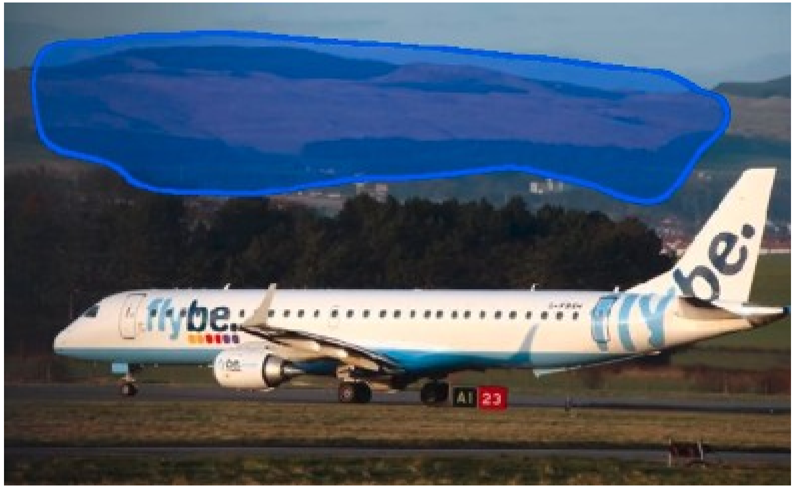
- **Project output:**
  - Machine learning pipeline that compares pre-trained deep learning models for different computer vision tasks.
  - Report with comparison of large pre-trained computer vision models and their properities in different machine learning tasks.



- **References:**
  - ImageNet Classification with Deep Convolutional Neural Networks
  - Deep Residual Learning for Image Recognition
  - Going deeper with convolutions
  - https://tfhub.dev/

Simultaneous speech and object recognition

- **Problem definition** – developing a system that learns to identify objects within an image, based on a spoken description of the image. Given an image and an audio caption, the model will highlight in real-time the relevant regions of the image being described.”
- **Examples:**
  - Jointly Discovering Visual Objects and Spoken Words from Raw Sensory Input
  - https://vimeo.com/user89645649
- **Expected output:**
  - Replicating results from the paper
  - Preparing a simple DEMO



Contact us:

http://tcl-research.pl

bartosz.biskupski@tcl.com