Easter European Machine Learning Summer School Reviews

Aug 10, Jeong Hyun Jae, Journal Club

Hyunjae Jeong (tAILab, CCIDS)

Yonsei University, Medical Life Systems Information Center (TAIL Lab)
Severance Hospital, Center for Clinical Imaging Data Science (CCIDS)
Severance Hospital, Radiology

CCIDS 의료영상데이터사이언스센터 Center for Clinical Imaging Data Science

tAILab.

01 Introduction

Doina Precup

McGill University

DeepMind

Ferenc Huszár

University of Cambridge

Razvan Pascanu

DeepMind

Viorica Patraucean

DeepMind

Dovydas Čeilutka

Vinted

Jev Gamper

Vinted

Karolina Dziugaite

McGill University

Google

Linas Baltrunas

Wayfair

Linas Petkevičius

Vilnius University

Neurotechnology

Schedule at a glance (Lithuania time)

The first 3 days (6–8 July) will be fully online. The last 4 days (11 - 14 July) will be hybrid; lectures and lab sessions will be held at the University of Vilnius, Faculty of Mathematics and Informatics, and streamed online for remote participants.

Wednesday 6	Thursday 7	Friday 8	Weekend 9,10	Monday 11	Tuesday 12	Wednesday 13	Thursday 14
10:00 - 10:15 Good morning							
AI Association Lithuania	Center of Pathology	MRU		Vilnius University	VilniusTech	KTU AI center	VDU
10:20 - 11:50 Intro DL (Razvan)	10:20 - 11:30 Fairness (Moritz)	10:20 - 11:30 Theory of DL (Matus)		10:20 - 11:30 Causality (Alexandre)	10:20 - 11:30 GraphNets (Thomas)	10:20 - 11:30 Journal club #B	10:20 - 11:30 Generalisation (Karolina)
	11:30 - 11:40 Break	11:30 - 11:40 Break		11:30 - 12:00 Industry Keynote	11:30 - 12:00 Industry Keynote	11:30 - 12:00 Industry Keynote	11:30 - 12:00 Industry Keynote
11:50 - 12:00 Break	11:40 - 13:10 Posters #B Sponsor Booths	11:40 - 13:10 Posters #C Sponsor Booths		12:00 - 12:20 Coffee Break			
12:00 - 13:10 Vision Models (Victor)				12:20 - 13:30 NLP (Harm)	12:20 - 13:30 Best practices in ML research (panel)	12:20 - 13:30 ML research in Lithuania (panel)	12:20 - 13:30 Project presentations
13:10 - 15:00 Lunch, Mentorship, Project discussions				13:30 - 15:00 Lunch, Mentorship, Project discussions, Onsite sponsor booths			
15:00 - 16:10 ML for speech (Tara)	15:00 - 16:10 Deep RL (Doina)	15:00 - 16:10 ML for drug discovery (Hannes)		15:00 - 16:10 Interpretability (Finale)	15:00 - 16:10 Journal club #A	15:00 - 16:10 RL (Michal)	15:00 - 16:10 Bayesian DL (Yee Whye)
16:10 - 16:30 Break				16:10 - 16:30 Coffee Break			
16:30 - 18:00 Explainability in Deep Learning tutorial	16:30 - 18:00 RL tutorial (Diana, Feryal) razvan	16:30 - 18:00 Career advice (panel)		16:30 - 18:00 Causality tutorial (Alexandre)	16:30 - 17:40 Diffusion talk (Aditya)	16:30 - 18:00 GraphNets tutorial (Catalina, Iulia)	16:30 - 17:00 Closing notes
18:00 - 19:00 Posters #A Sponsor Booths	18:00 - 19:00 Theory of DL (Suryia)						

01 Introduction

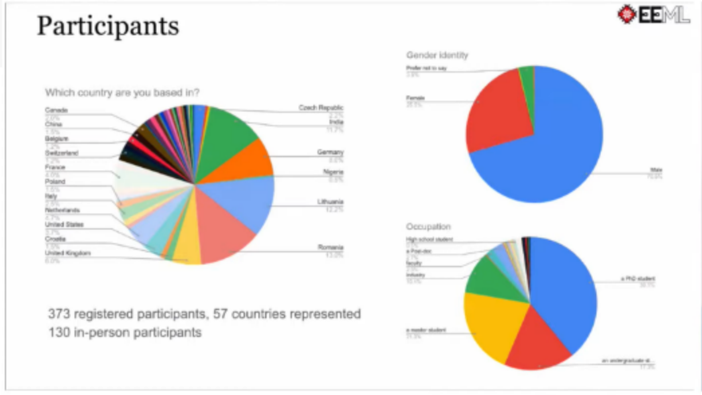
tAILab.

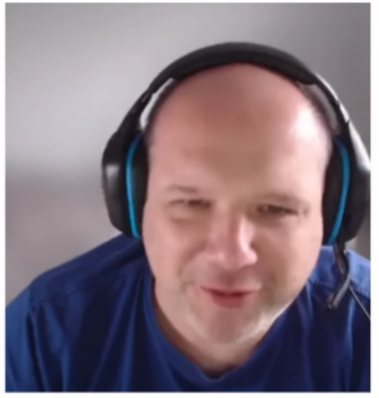
Introduction to Deep Learning

Introduction to Deep Learning

Razvan Pascanu / DeepMind

In this talk I will attempt to provide an introduction to deep learning. The goal is to cover most of the important concepts, providing everyone a similar starting point for future lectures, but also to discuss some of the main open issues in the field. I will cover basics from backpropagation and gradient descent, to describing the main insights that led to architectures like transformers, graph nets, LSTMs or ResNets. And try to highlight the implicit assumptions, limitations, and exciting new directions. The lecture aims to have something for everyone, from beginners to those familiar with the topic.





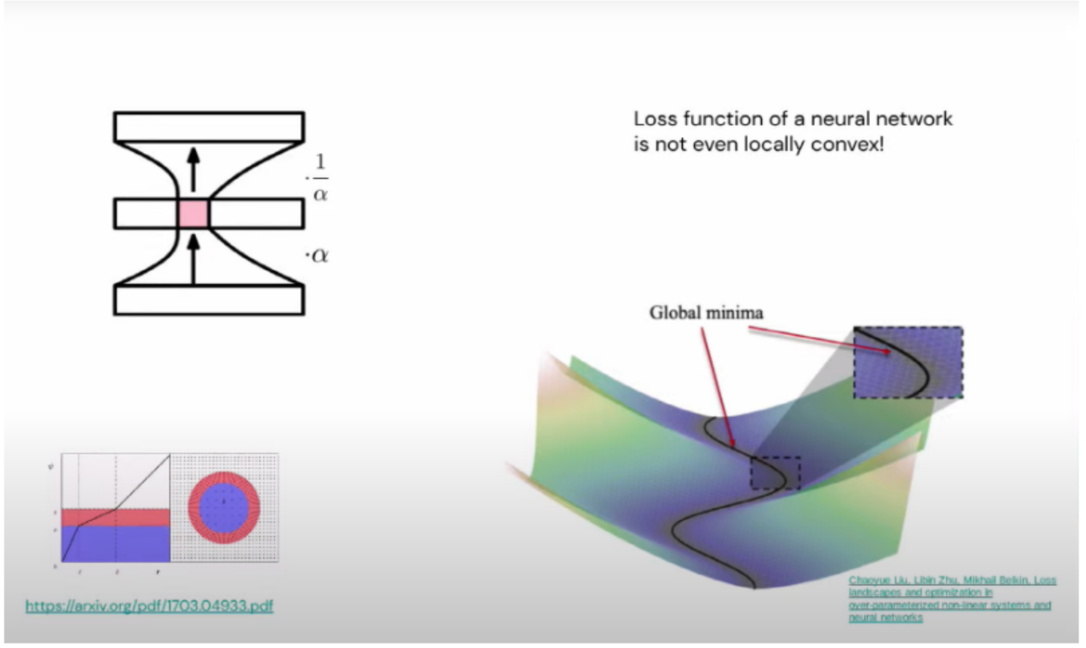
The stream is sponsored by Vinted

EEML

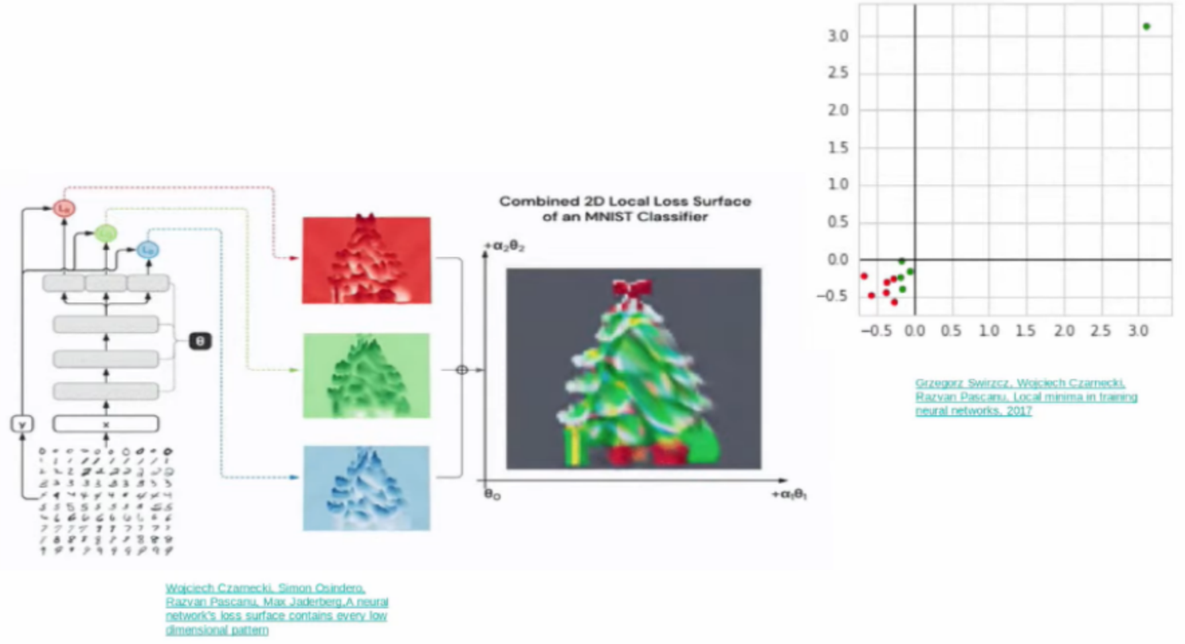
tAILab.

Supervised & Parametric Models

- We should focused on two questions : 1. What is a good parametrization? 2. How do we find these optimal par ameters? + Inductive Bias Problem
- The Deep Learning Myth :Deep Networks can be inserted almost anywhere and will just work Optimization is as if the loss was convex -> Sapple Points or not even locally convex !



Deep Learning Myth



MNIST Loss Function Surfaces

tAILab.

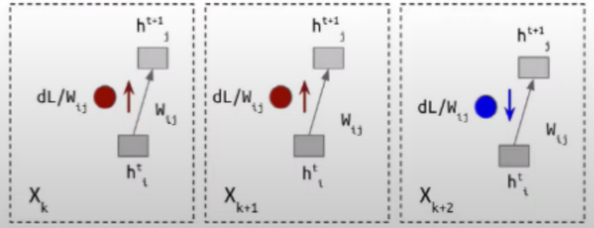
Backpropagation Algorithm

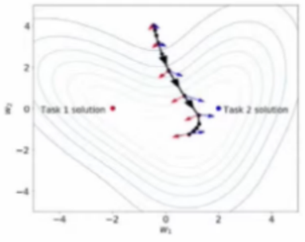
- Optimization Functions -> SGD, Momentum, Adagrad ...
- How is it different from the chain rule?
- Gradient Descent = Tug of war game ! , Regularization = Bulk Up Guy
  This requires data to be I.I.D, no explicit knowledge composition

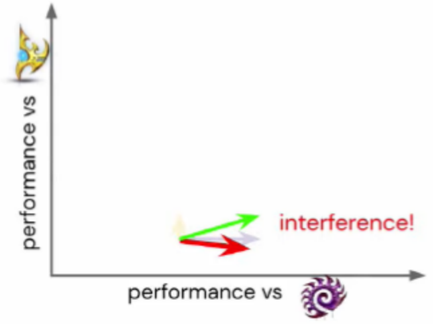
The IID assumption in Gradient Descent

- **Independently for each data point and parameter** ask how much the loss would change if a parameter is increased or decreased in magnitude
- Modify the parameter by taking **a small step** proportional with the impact it has on the loss

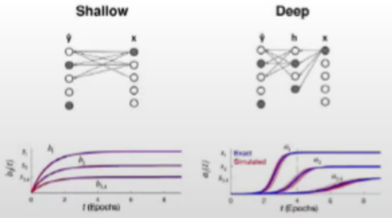
$$\frac{\partial L}{\partial W_{ij}} = \lim_{\epsilon \to 0} \frac{L(W_{ij}) - L(W_{ij} + \epsilon)}{\epsilon}$$







Tom Schaul et al 2019, Ray interference: a source of Plateaus in Deep Reinforcement Learning

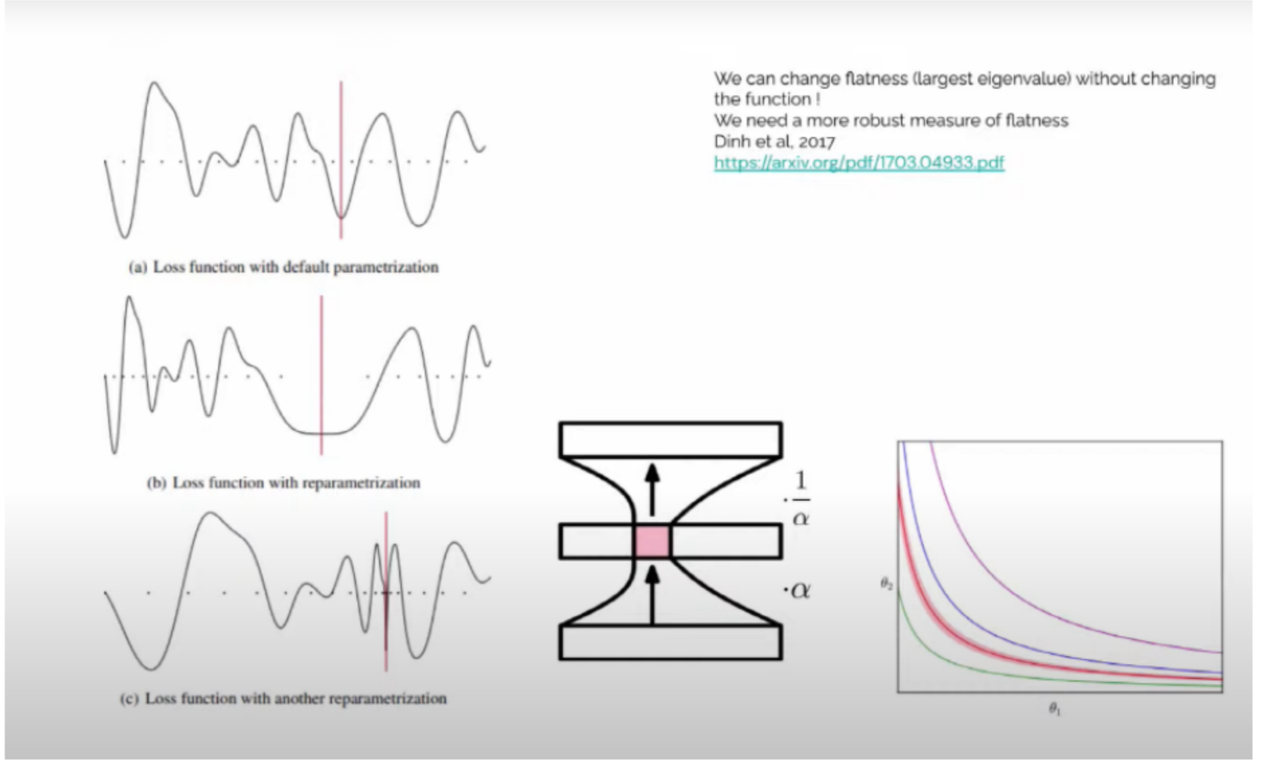


Andrew Saxe et al 2013,Exact solutions to the nonlinear dynamics in deep linear models

tAILab.

Inductive Bias - Powerpropagation, Reparametrization

- restrict or reshape the search space for the optimal parameters : powerpropagation, reparametrization
  - e.g. SAM optimizer
- Powerpropagation = A new weight-parameterisation for NN



Powerpropagation:
A sparsity inducing weight reparameterisation

Jonathan Schwarz
DeepMind &
Gatsby Unit, UCL
schwarzjn@google.com

Siddhant M. Jayakumar
DeepMind &
University College London

Razvan Pascanu
DeepMind

Peter E. Latham
Gatsby Unit, UCL

Yee Whye Teh
DeepMind

Abstract

The training of sparse neural networks is becoming an increasingly important tool for reducing the computational footprint of models at training and evaluation, as well enabling the effective scaling up of models. Whereas much work over the years has been dedicated to specialised pruning techniques, little attention has been paid to the inherent effect of gradient based training on model sparsity. In this work, we introduce Powerpropagation, a new weight-parameterisation for neural networks that leads to *inherently sparse* models. Exploiting the behaviour of gradient descent, our method gives rise to weight updates exhibiting a "rich get richer" dynamic, leaving low-magnitude parameters largely unaffected by learning. Models trained in this manner exhibit similar performance, but have a distribution with markedly higher density at zero, allowing more parameters to be pruned safely. Powerpropagation is general, intuitive, cheap and straight-forward to implement and can readily be combined with various other techniques. To highlight its versatility, we explore it in two very different settings: Firstly, following a recent line of work, we investigate its effect on sparse training for resource-constrained settings. Here, we combine Powerpropagation with a traditional weight-pruning technique as well as recent state-of-the-art sparse-to-sparse algorithms, showing superior performance on the ImageNet benchmark. Secondly, we advocate the use of sparsity in overcoming catastrophic forgetting, where compressed representations allow accommodating a large number of tasks at fixed model capacity. In all cases our reparameterisation considerably increases the efficacy of the off-the-shelf methods.

1 Introduction

Deep learning models are emerging as the dominant choice across several domains, from language [e.g. 1, 2] to vision [e.g. 3, 4] to RL [e.g. 5, 6]. One particular characteristic of these architectures is that they perform optimally in the overparameterised regime. In fact, their size seems to be mostly limited by hardware constraints. While this is potentially counter-intuitive, given a classical view on overfitting, the current understanding is that model size tends to have a dual role: It leads to better behaved loss surfaces, making optimisation easy, but also acts as a regulariser. This gives rise to the

01 Introduction

tAILab.

Introduction to differentiable scene representations and learn able 3D reconstructions

Introduction to differentiable scene representations and learnable 3D reconstruction

Victor Lempitsky /

Deep learning and differentiable rendering are now widely used to reconstruct 3D scenes from single images, multiple images or videos. To perform learnable reconstruction, 3D scenes need to be represented in a differentiable manner. This can be done in several rather different ways, and each of the ways lead to interesting algorithms and reconstruction results. In the lecture, I will give a brief overview of the recently introduced scene representations, their pros and cons, and how these representations are employed within 3D reconstruction algorithms

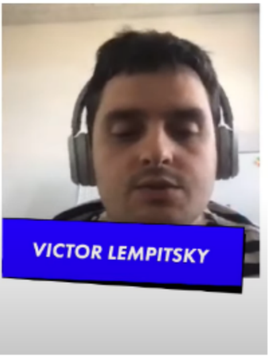
Geometry + appearance decomposition

Certain model of geometry: mesh, point cloud, volumetric density

Certain model of *appearance*: which radiance is observed at non-transparent points

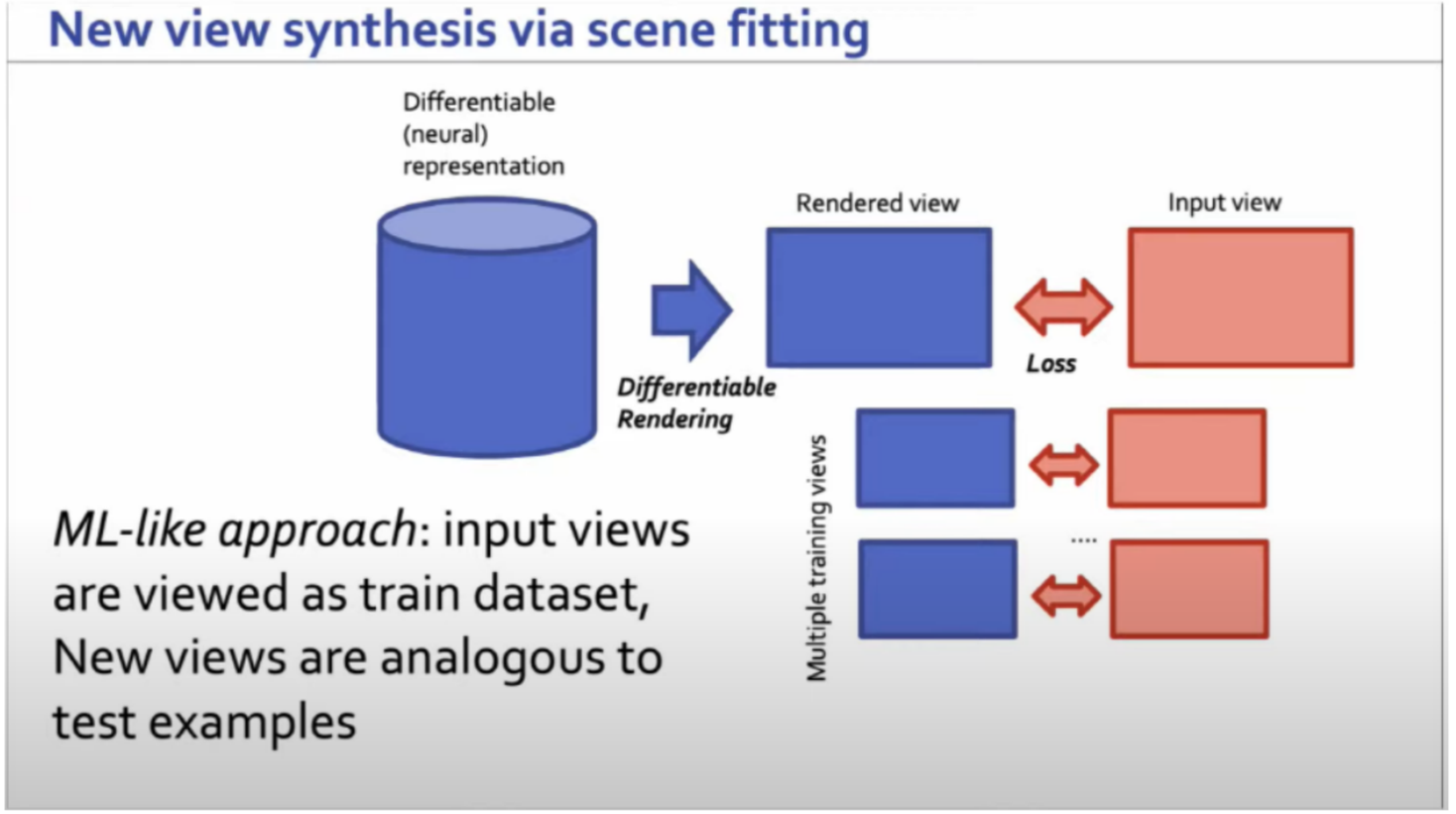
Radiance does not change along the ray, if there are no occluders (no surfaces, no opacity)

Plenoptic function can then be computed by ray marching



New view synthesis via scene fitting

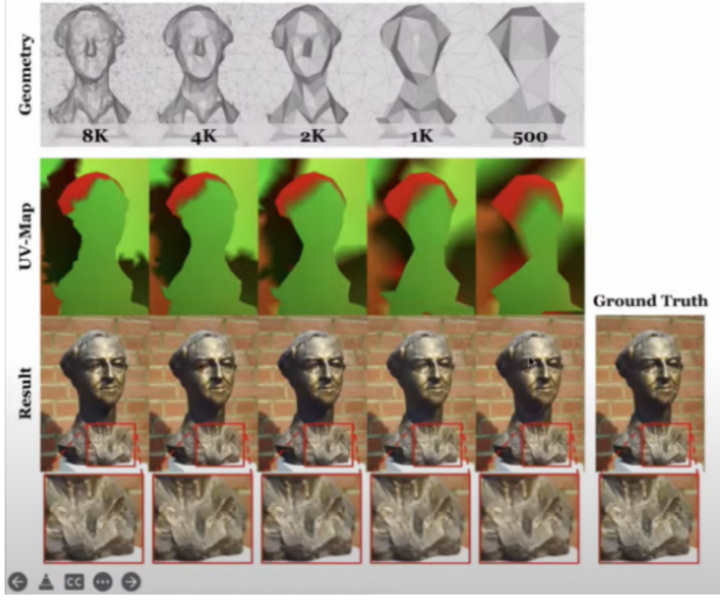
- To perform learnable reconstruction, 3D scenes need to be represented in a differentiable manner.
- There are two main types of fitting loss: pixelwise difference between input and rendering and 형태 유지 loss.



Neural Rendering - improving mesh+texture with a NN

- To perform learnable reconstruction, 3D scenes need to be represented in a differentiable manner.
- Deferred neural rendering -> Fitting to a scene using mesh geometry + neural texture + rendering net, neural r endering network is used as the last stage of the pipeline -> resolution does not affect prediction quality!!!
- Neural dressing model: SMPL-X body model - Motion excluding hair and clothes, + Neural texture can be added to the body model to obtain the result of changing clothes and hair. Here textures can be found anywhere. (Gri gorev et al.CVPR 21))

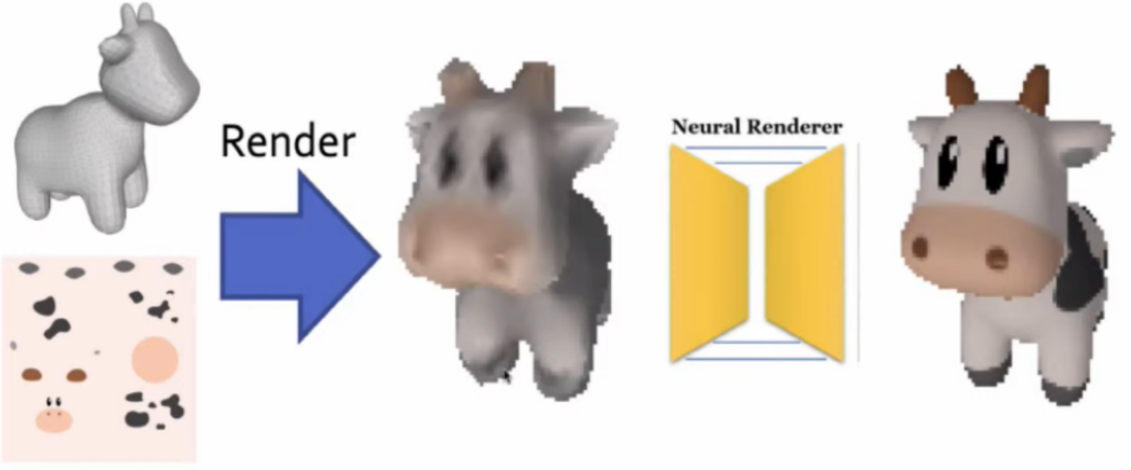
Deferred neural rendering



- Typical number of channels = 8 or 16
- Additional input channels encode view direction
- Realistic images are generated even for coarse geometry!

[Thies et al. ACM ToG 2019]

Improving mesh + texture representation

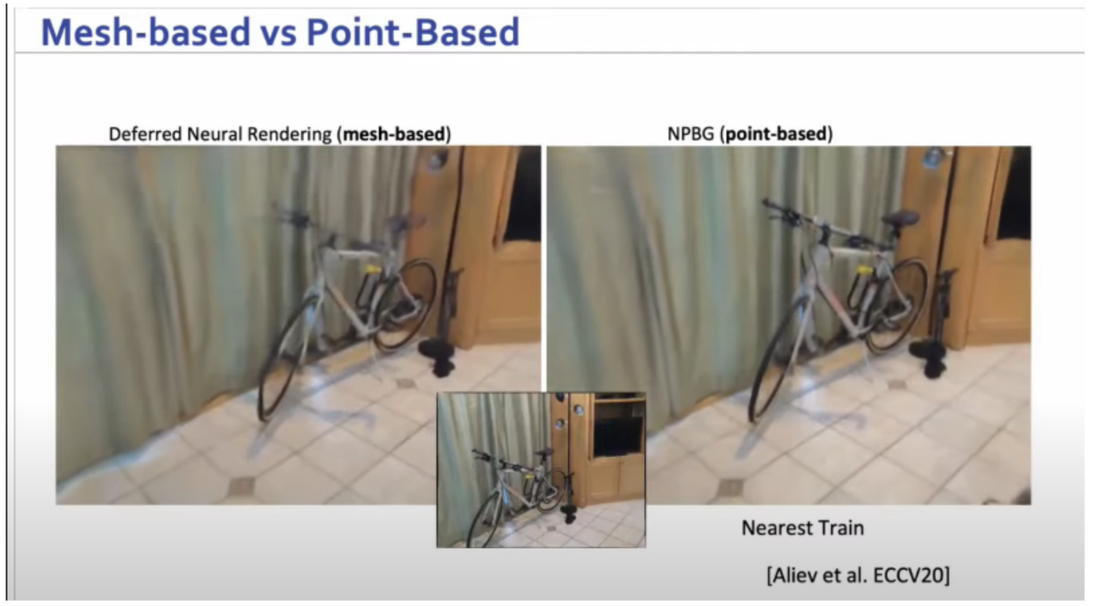


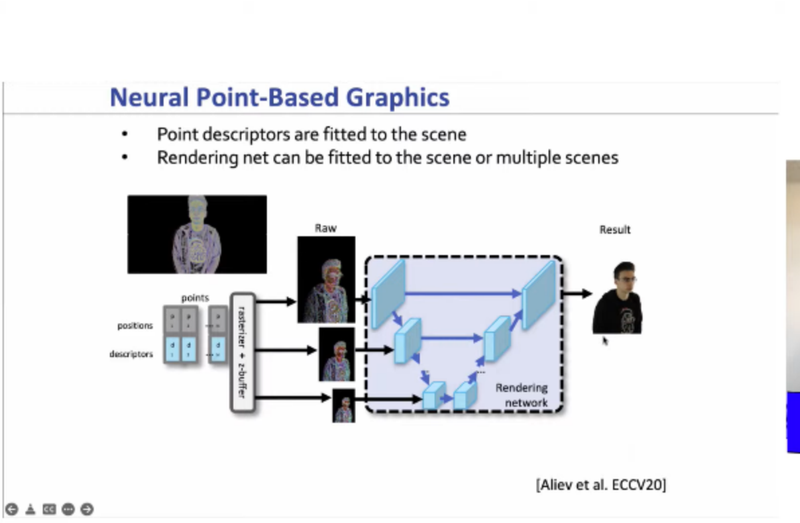
tAILab.

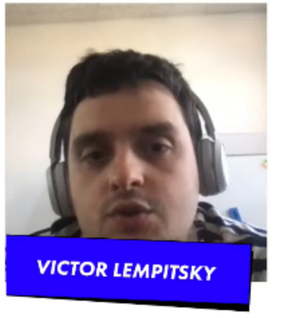
Neural Point-Based Graphics

- Point Cloud -> point decriptors are fitted to the scene, rendering net can be fitted to the scene or multiple scenes

- NeRF (Neural Radiance Fields), MIP NeRF, NeRF baking, Direct Voxel-Grid Optimization, Signed distance function SDF + NeRF







EEML

01 Introduction

tAILab.

Advances in Digital Pathology

Unlimited set of tasks and solutions

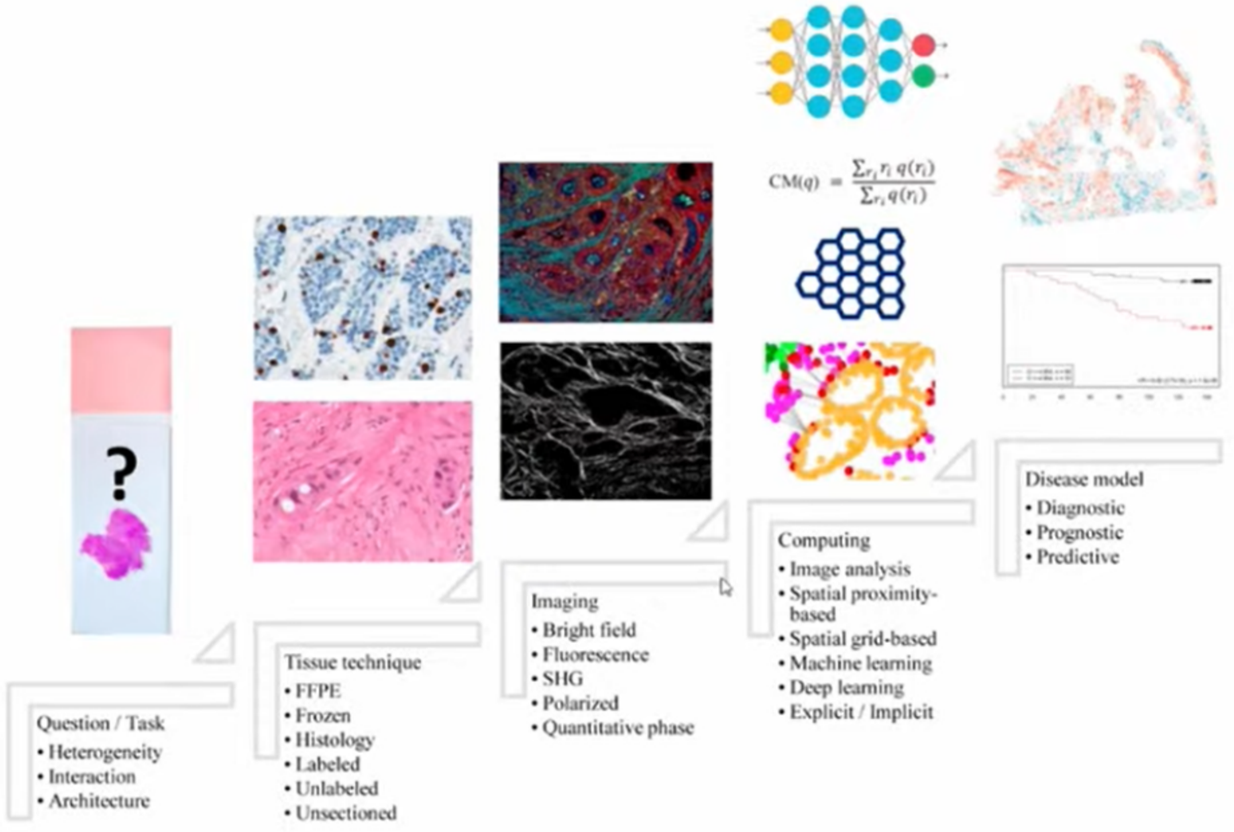
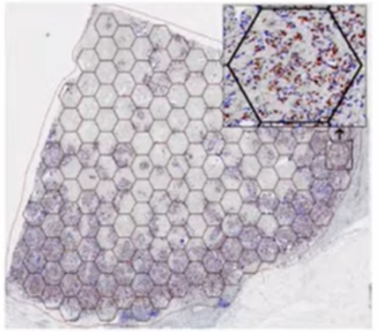
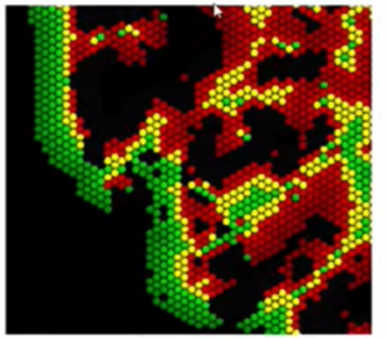


Image-based, computational biomarkers/models

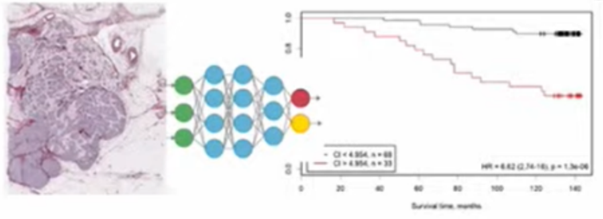
1. Intratumor Heterogeneity



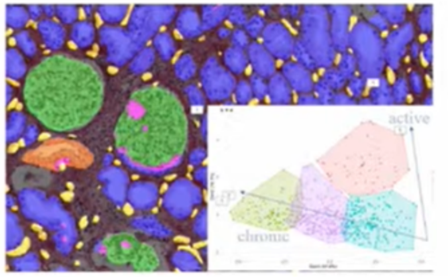
2. Tumor-Host Interaction



3. Histology-based deep learning

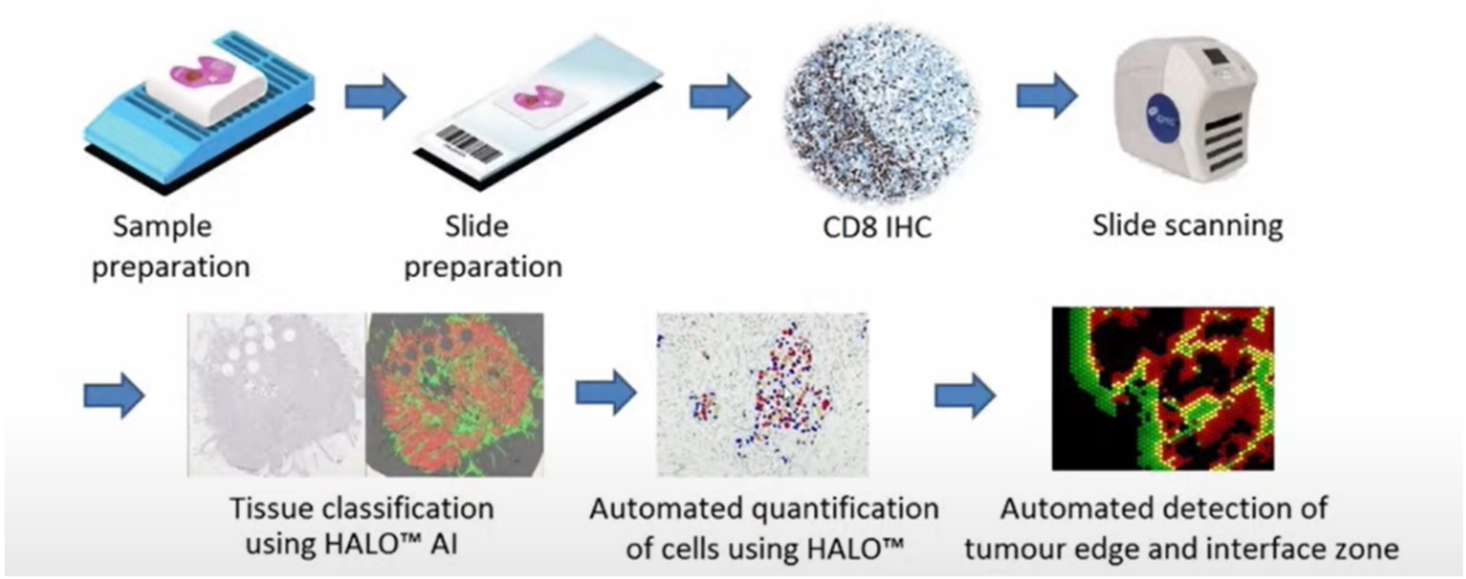


4. Multivariate Kidney Morphometrics

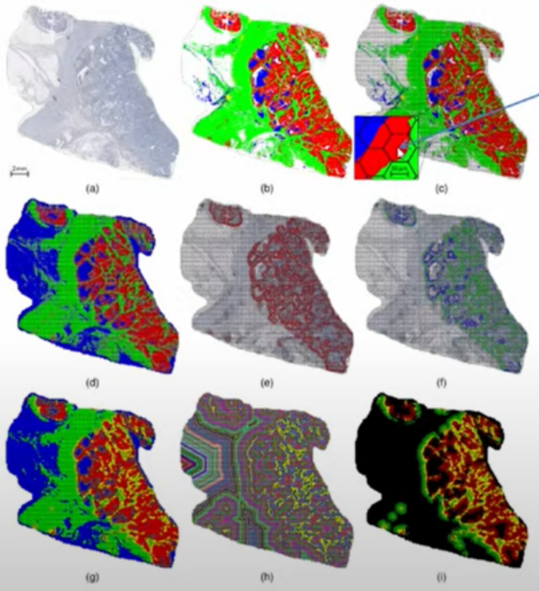


01 Introduction

Tumor-Host Interaction

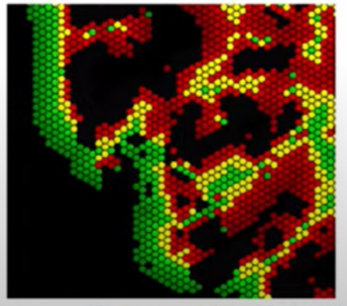


Extracting Tumor Edge and Interface Zone



1. Probability of "interface-ness" to detect **Tumor Edge**

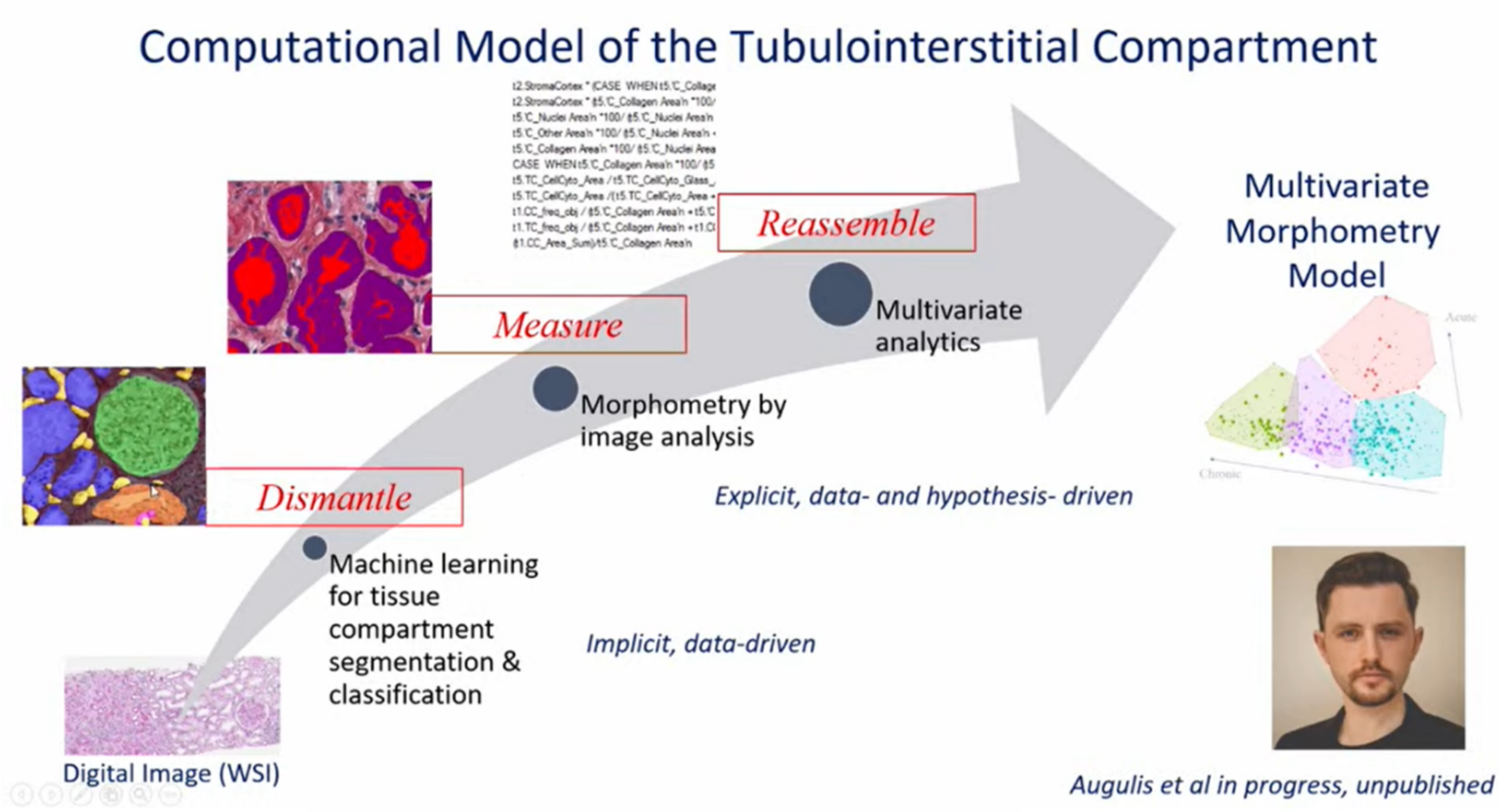
2. Spatially ranked **Interface Zone** around the **Tumor Edge**



Rasmusson et al Am J Path 2020

01 Introduction

Multivariate Kidney Morphometrics



Augulis et al in progress, unpublished

tAILab.

End-to-End Speech Recognition: The Journey from Research to Production

End-to-End Speech Recognition: The Journey from Research to Production

Tara Sainath / Google Research

End-to-end (E2E) speech recognition has become a popular research paradigm in recent years, allowing the modular components of a conventional speech recognition system (acoustic model, pronunciation model, language model), to be replaced by one neural network. In this talk, we will discuss a multi-year research journey of E2E modeling for speech recognition at Google. This journey has resulted in E2E models that can surpass the performance of conventional models across many different quality and latency metrics, as well as the productionization of E2E models for Pixel 4, 5 and 6 phones. We will also touch upon future research efforts with E2E models, including multi-lingual speech recognition.

TEAMS ›

Brain Team

Make machines intelligent. Improve people's lives.

About the team

History of Research Breakthroughs

Google Brain started in 2011 at X as an exploratory lab and was founded by Jeff Dean, Greg Corrado and Andrew Ng, along with other engineers and is now part of Google Research. Since then, we continually rethink our approach to machine learning and are proud of our breakthroughs, which include:

- AI infrastructure (developing TensorFlow)
- Sequence-to-sequence learning, leading to Transformers and BERT
- AutoML, pioneering automated machine learning for production use

Our research breakthroughs enable Google's mission to organize the world's information and make it universally accessible and useful.

Google Impact

As part of Google and Alphabet, the Brain team has access to resources and unparalleled collaboration opportunities that have a positive impact on products and society. Our broad and fundamental research goals allow us to collaborate with and contribute to many teams across Alphabet, which deploy our cutting-edge technology into products used by billions of users, positively impacting society and the research community.

Open and Bottom-Up Culture

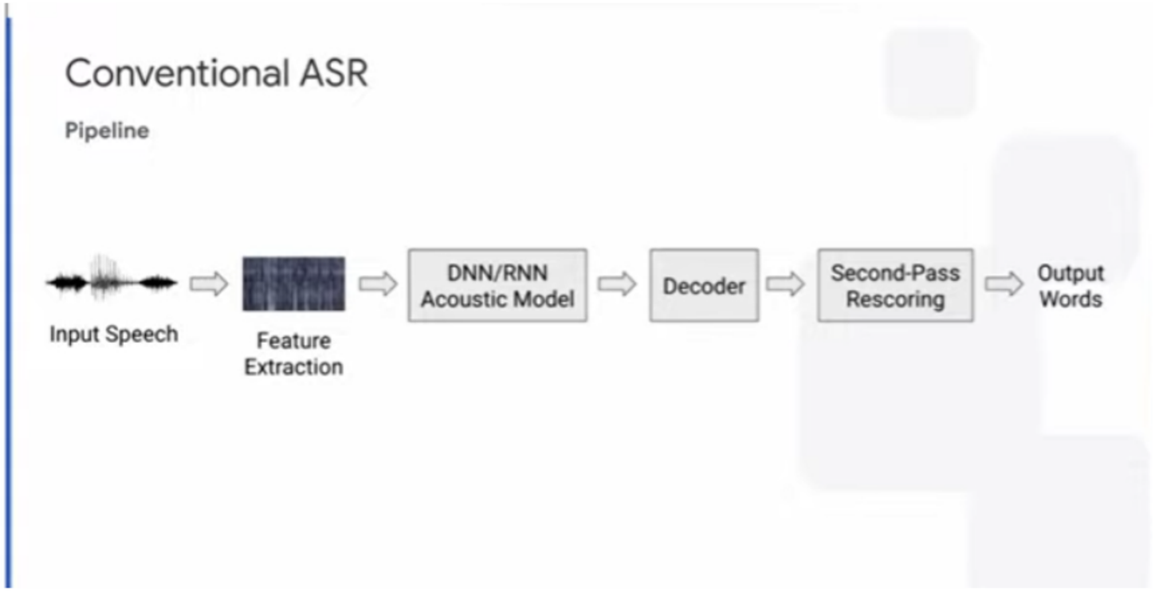
We believe that openly disseminating research is critical to a healthy exchange of ideas, leading to rapid progress in the field.

As such, we regularly publish our research at top academic conferences and journals, and release our tools, such as TensorFlow and Jax, as open-source projects.

Team members are encouraged to set their own research goals, allowing the Brain team to maintain a portfolio of projects across varied time horizons, research areas and levels of risk.

What is End-to-End ASR

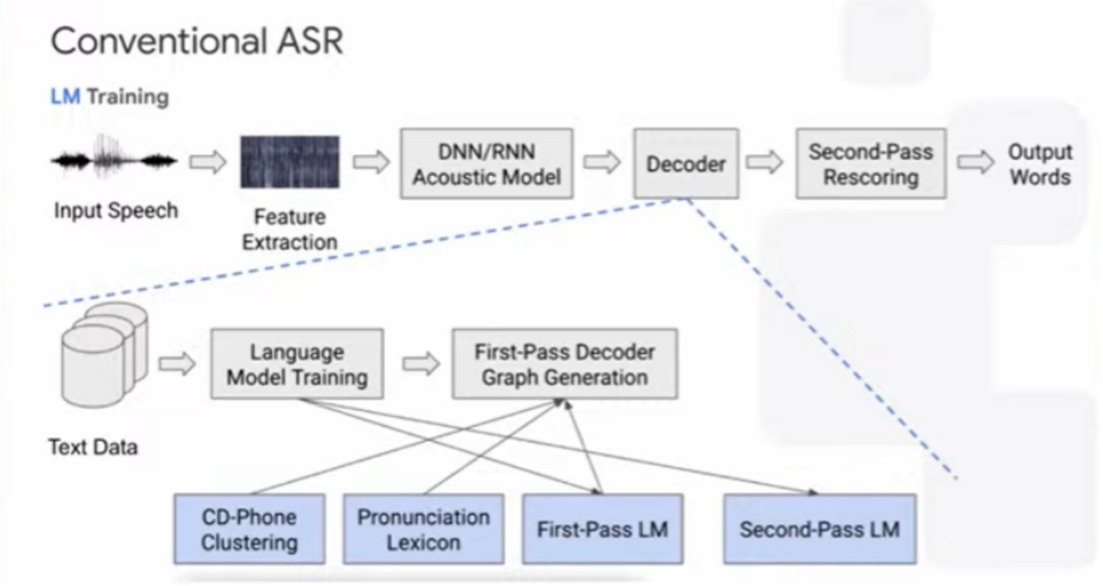
- ASR = Automatic Speech Recognition
- A system which is trained to optimize criteria that are related to the final evaluation metric - word error rate
- End-to-End : A system which directly maps a sequence of input acoustic features into a sequence of graphemes or words
- Conventional ASR Pipeline

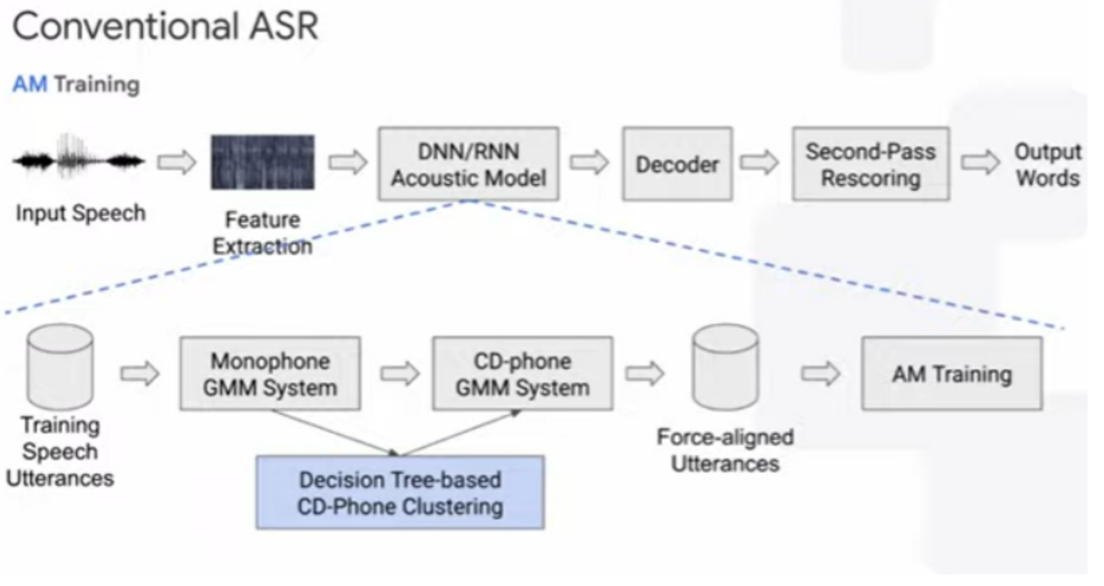


tAILab.

What is End-to-End ASR

- Combine Monophone and CD-phone. Because each device is different, Domain Specific Acoustic knowledge is Complicate
- End-to-End : A system which directly maps a sequence of input acoustic features into a sequence of grapheme s or words
- LM Training & AM Training
  LM = Language Model, AM = Acoustic Model





Historical Development of End-to-End ASR

1. Connectionist Temporal Classification (2006 ICML) : CTC allows for trainin g an acoustic model without the need for frame-level alignments between the acoustics and the transcripts

2. Attention-based Encoder-Decoder models (2015)

3. Comparing Various End-to-End Approaches (2017) -> online/offline

4. Further Improvements (2018)

4 - 1 Multi-head Attention : Multi-headed attention examines different pa rts of the utterance for each predicted label, model looks predominantly to wards previous frames

4 - 2 Word Piece : Longer units have a lower LM perplexity, Longer units gives improved decoder efficiency.

4 - 3 Minimum Word Error Rate (MWER)

4 - 4 Comparison to Conventional Model : Decreasing Error Rate and mod el size. Streaming is critical to production

Comparing Various End-to-End Approaches
[Prabhavalkar et al., 2017]

Model	Online/Offline	VoiceSearch Word Error Rate (%)
Conventional Model	online	9.9
	offline	8.6
CTC-Grapheme (no LM)	online	53.4
Attention-based Model	offline	11.7

- Decoding CTC-grapheme models without an LM performs poorly.
- Attention-based model performs better, but still lags behind a conventional model.

tAILab.

Historical Development of End-to-End ASR (2019-2020)

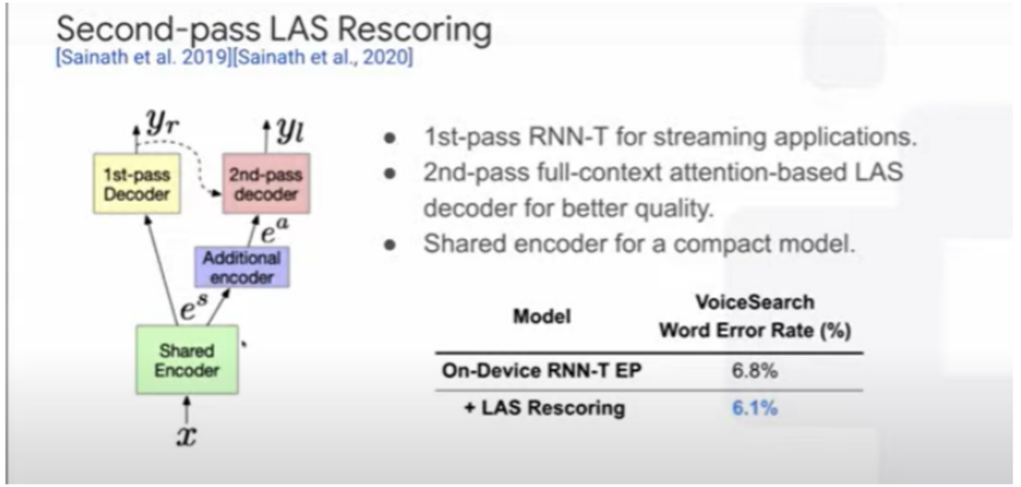
5. Streaming speech recognition -> Recognize Speech ASAP.
Online End-to-End System

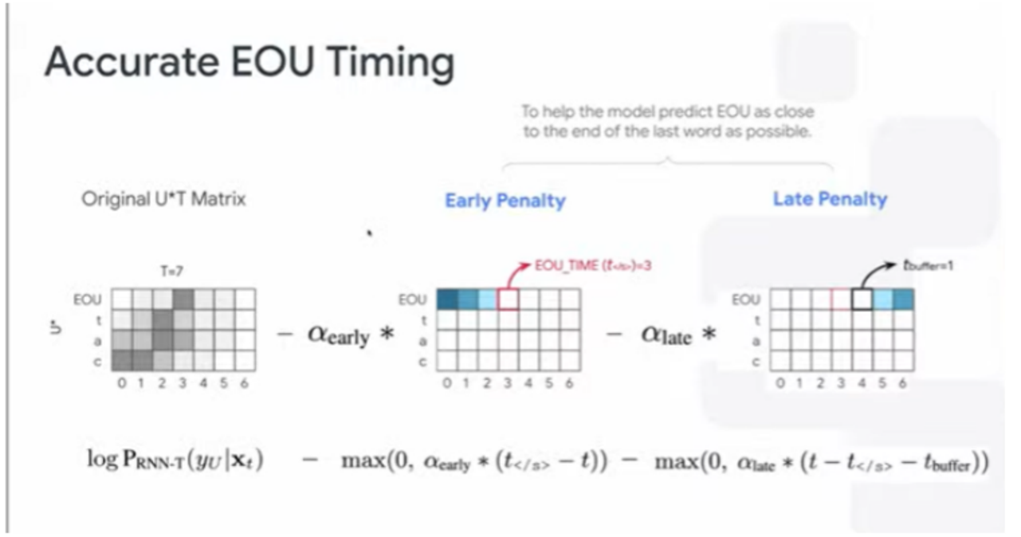
5 - 1 Recurrent Neural Network Transducer (RNN-T) : Prediction Network와 Encoder 두 개로 나뉜다. t = 0부터 T까지의 frame을 설정한다. 그래서 frame당 단어를 측정하는 것. 그래서 결국 마지막에 어떻게 Terminate 할래? -> End-Point = EOU. 아주작은 neural network로 만들어져 있는데, 침묵을 측정한다. Semantic으로 알 수 있다. (Low Latency RNN-T, also called RNN-T Endpointer) 어떻게 할래? -> Penalize Early End and Late Penalty (Matrix of U and T)

5 - 2 How to maintain Quality when streaming -> Second-pass LAS Rescoring.

6. Combining all of this is the Google voice recognition model used in 2020. (Pixel4) -> Surpassing, much better quality and latency -> much faster, less rely on google server) -> how quick and how accurate!

- “You could tell how amazed the crowd back in 2019 was about the "lyft ride to hotel" example in the tech demo, I guess since it also incorporated some other inferences like "my hotel" and all of this is so quickly that one would prefer doing it rather than by yourself. Pretty cool!”





Historical Development of End-to-End ASR (2021 Now)

2021 On-Device Improvements (Pixel 6)

1. Surpasses quality (word error late)

2. Faster in terms of latency (endpointer, computational)

3. Lower power
-> Pixel 6 Google Tensor SoC Hardware

6 - 1 Latency Improvement : LSTM Encoder -> Conformer Encoder (LSTM -> Time dependency and gr adient vanishing, not TPU friendly -> because Sequential time dependency) Latency Improvements wi thout quality drop + 30% computation speed up with no accuracy degradation.

6 - 2 Quality Improvement : Multi-rate Encoders -> Too expensive to beam search in device, Quality Im provement : Neural Language Model -> Hybrid Autoregressive Transducer (HAT) factorization to bette r integrate language model - Video