
Deep Learning (Fall 2023)

# **Mid-Term Presentation** | Merging Pixels and Words: A VQA System for Remote Sensing

Vital Visions - Kaan Aydin, Damian Falk

# Agenda

- Recap of our Project
- Details on Training Setup
- Experiments & Initial Results
- Path Forward & Status

# Recap | Problem Statement…

Advancements in sensing technology have led to a **large volume of visual info** being acquired and processed in the Remote Sensing (RS) domain

RS data has **immense potential for various tasks** (e.g., land cover classification, object detection) which could benefit numerous applications

Current methods are **static and task-specific** and require specialized expert knowledge – hindering generic and easy access to the information

# … and our proposed solution

Remote Sensing Visual Question Answering System (RSVQA) allows for **info extraction via queries** formulated in natural language

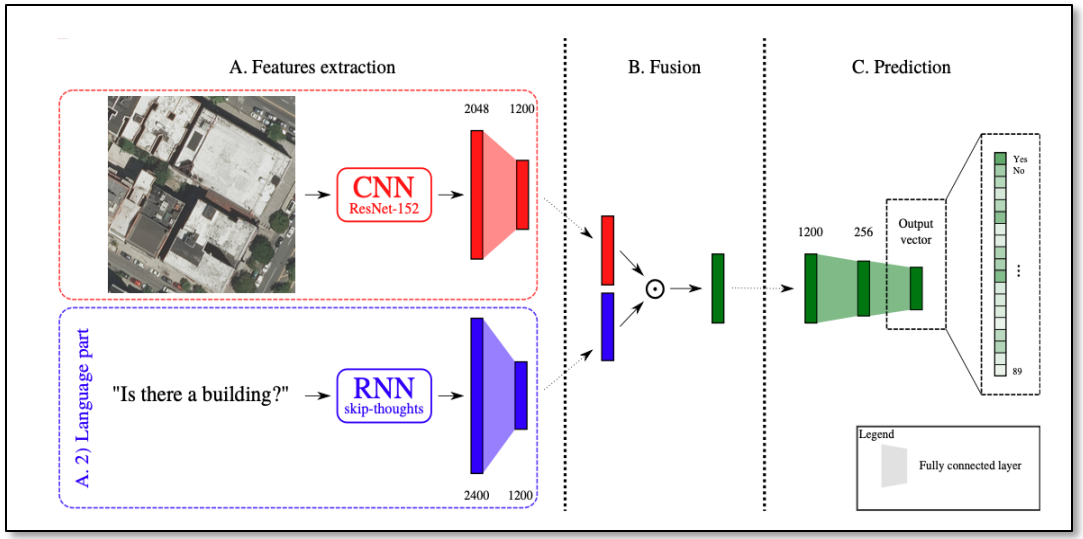
VQA enables **answering of free-form and open-ended questions -** providing a more intuitive solution to data extraction / analysis

This approach can **address problems** such as:

- Specific object detection (e.g., "Is there a thatched roof in this image?")
- Complex tasks, e.g., relations btw. different objects ("Is there a building on the right of the river?")

# **Recap** | We derive our approach from our blueprint paper

## Framework of the VQA model [1]




## Description

### A. Feature Extraction

- **Visual:** ResNet-152, pretrained on ImageNet
- **Language:** Skip-thoughts, pre-trained on BookCorpus
- FC layers to reduce dimensions to 1200

### B. Fusion

- Point-wise multiplication to each element
- Fixed operation – e2e training ensures comparability of features

### C. Prediction

- Projection of 1200 dimensional vector to answer space by using MLP
- Formulation of problem as a classification task



**Sources:** [1] Lobry, Marcos, Murray & Tuia (2020)

## RSVQA Dataset

- Dataset consists of one high- and one low-resolution dataset
- We decided to focus on the HR dataset given its larger size
- The questions are separated into four categories

~11k images

~1M questions

University of St.Gallen

# **Recap** | Overview of example image, question and ground truth answers for each category

### Area

What is the area covered by rectangular buildings?

Between 100 & 1000 $m^2$

### Presence

Is there a road in the image?

yes

### Count

How many buildings on the left of a road are there in the image?

38

### Comparison

Are there more residential buildings than roads?

yes

We started our project with a detailed review of the blueprint approach and **trained with standard configurations**

The standard training **took approximately 4.5 days**

As we wanted to evaluate multiple options, we first needed to **decrease time spent** for training

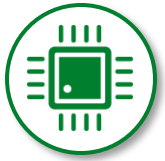
To do so, we focused mainly on **two levers**

- Preprocessing
- Training configurations

# **Training Setup** | We pre-computed the visual & textual embeddings to decrease training time

## Pre-Processing

- In the blueprint approach, the feature extractors are frozen during training
- Thus, instead of computing embeddings for each epoch, we pre-compute them
- With this setup, we can eliminate ~35M forward passes per training run and massively increase performance



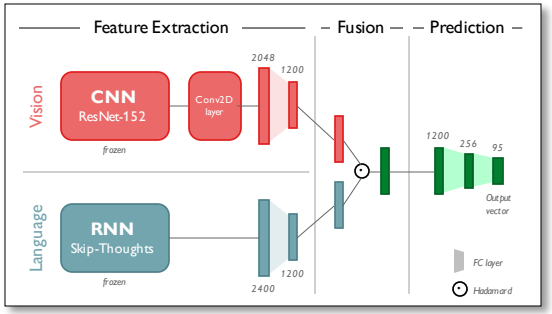
## Batch Size & Multi-Processing

- Due to a decreased memory footprint, we can scale our batch size by 10
- We also increased the number of workers to allow for faster I/O rates
- However, we realized that we could get similar speed with fewer workers by loading everything into memory on init

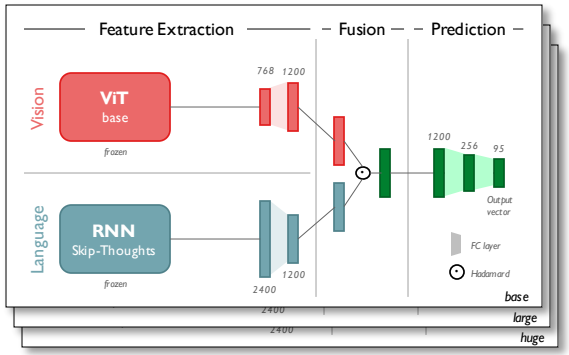
**These changes reduced our training from 4.5 days to 8 hours**
(and ultimately 4.5 hours with some further improvements)

# **Experiments** | So far, we ran three different experiment setups with varying model sizes & hyperparameters

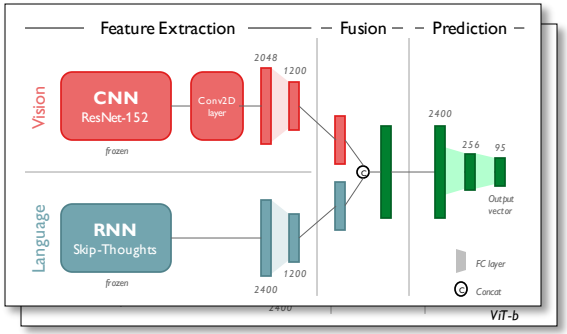
## **Experiment 1:** ResNet152 and Skip-Thoughts



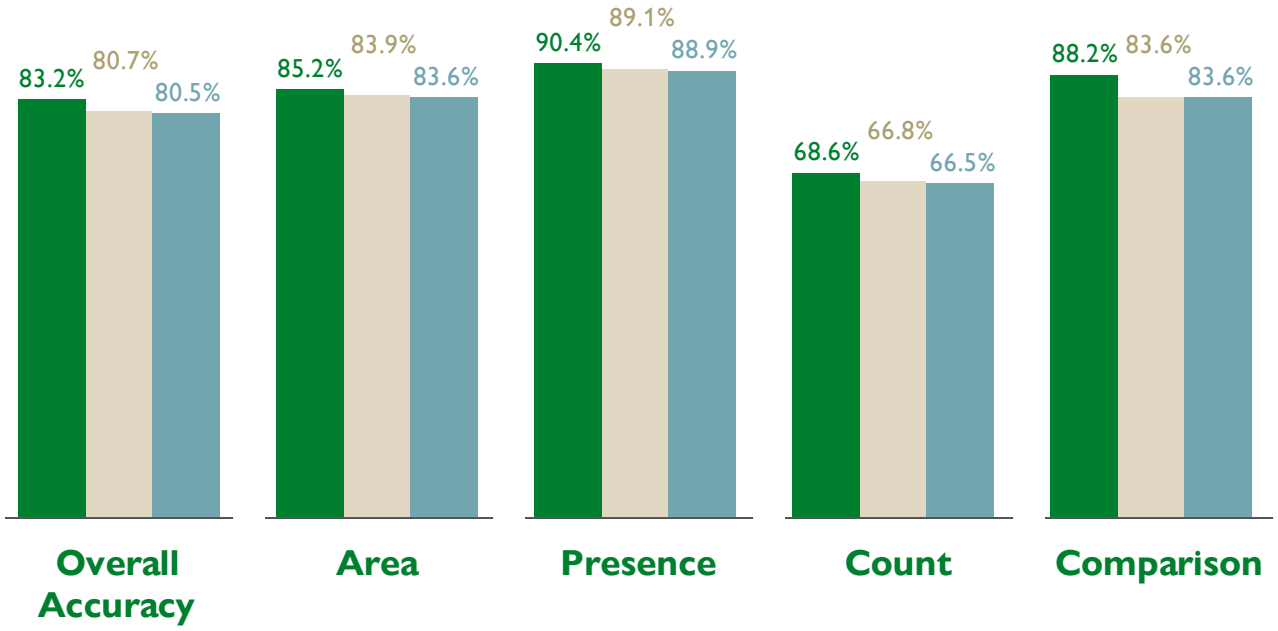

## **Experiment 2:** Vision Transformer and Skip-Thoughts




## **Experiment 3:** Fusion with concatenation



# **Initial Results** | Accuracy scores on the test set in Experiment 1 compared to blueprint paper

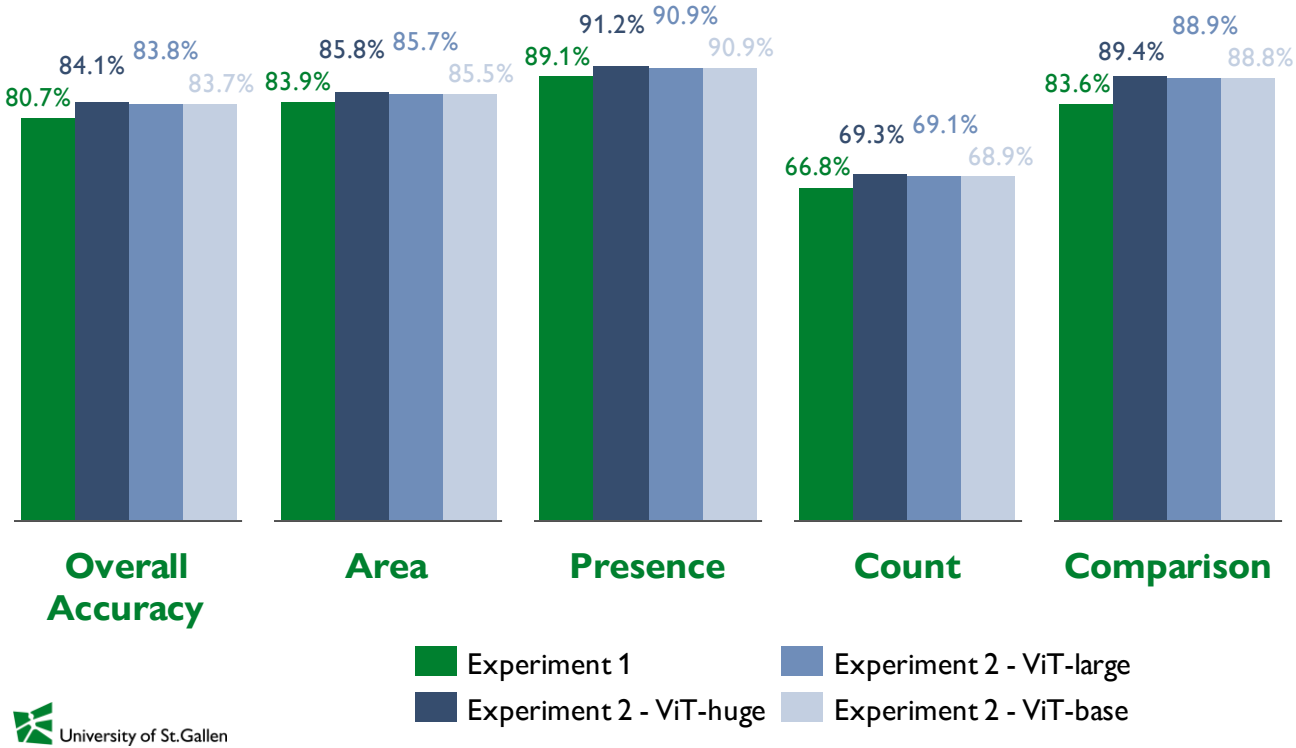



Overall, our training runs performed similarly to the results from the blueprint papers (~2 pc. points lower for Overall Accuracy)

We believe this is likely due to the initialization & randomization during training

Changes from pre-processing & training configs have small impact on accuracy

# **Initial Results** | Accuracy scores on the test set in Experiment 1 and 2



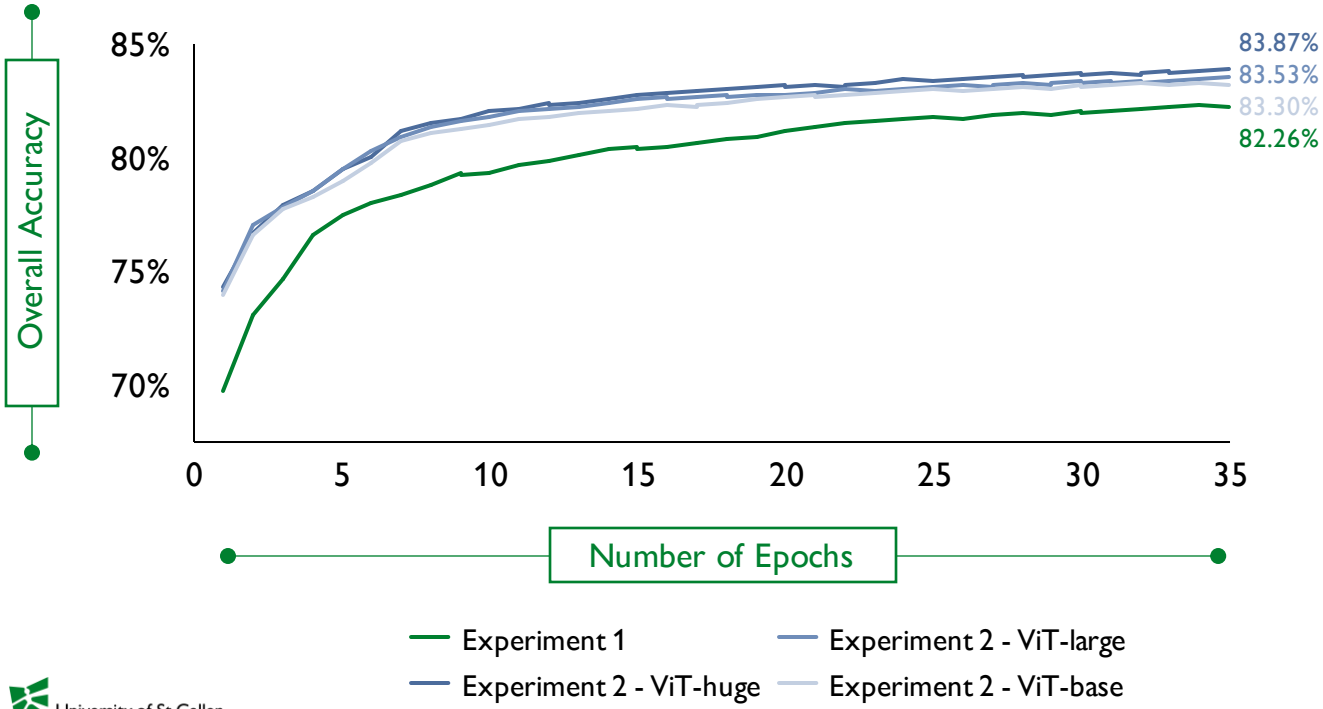

By using a ViT instead of a ResNet as feature extractor, we were able to improve the accuracy on all question types

This approach also improves accuracy compared to the blueprint paper in every category

Model size of the ViT on the other hand only shows a very small impact on performance

# **Initial Results** | Overall Accuracy over epoch on the validation set

*Loss over epochs in Appendix*

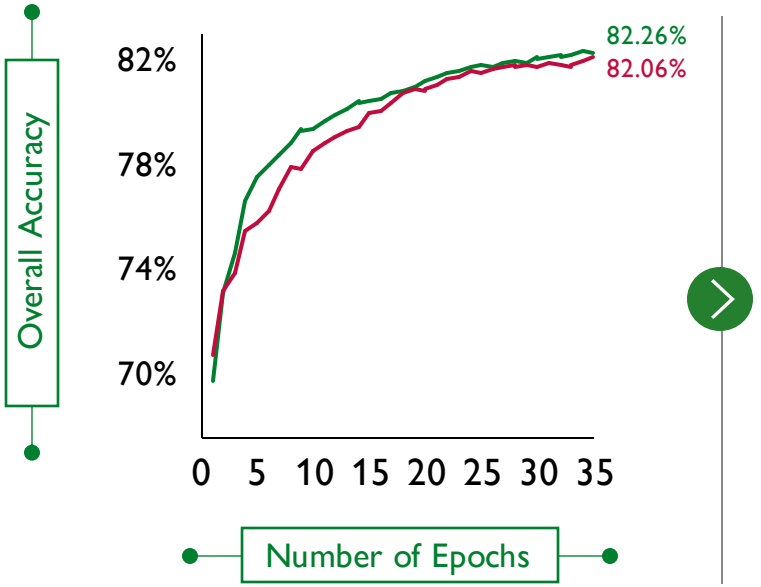



Looking closer at the accuracy over epochs, we see that the ViT starts with and consistently achieves a higher accuracy on the validation set

However, it still follows a similar trajectory and loses part of its advantage over time

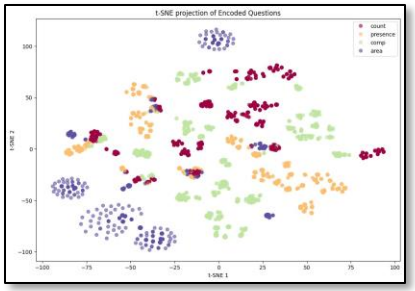
# **Initial Results** | Hadamard vs. concatenation

## Accuracy on Evaluation Set

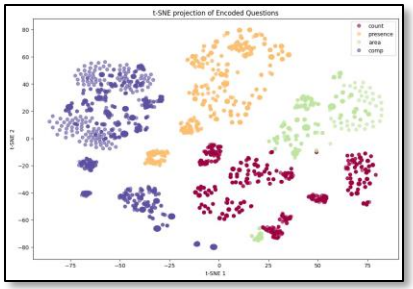



## t-SNE projections of encoded questions

with RNN




with BERT




Experiment 1 - Hadamard  Experiment 3 - Concat

Using the baseline set-up, concatenation and multiplication achieve similar scores after 35 epochs of training

Looking at the t-SNE projections of the question embeddings with the RNN we see that the question types are not well separated

Using BERT, the question embeddings look more disentangled which might help to achieve better performance

# Path Forward | We slightly adjusted our approach

As expected, we **slightly decided to deviate** from our suggested approach given new ideas, learnings & limitations

Initially, we had identified **three distinct areas** where we wanted to modify the blueprint approach:

- Other feature extraction methods (e.g., ViT, DETR)
- Pre-training methods
- Diverse data fusion approaches (e.g., MCB)

We exclude pre-training from our approach, but rather focus on:

- **Self and Cross-Attention** for contextualized features
- **Multi-Task Learning** to learn separate classifiers
- **Integration of BERT** for improved language feature extraction

**Status |** We completed most of our experiments – only two remaining to be finished

✓ *Done*

≈ *Still exploring*

## Feature Extraction

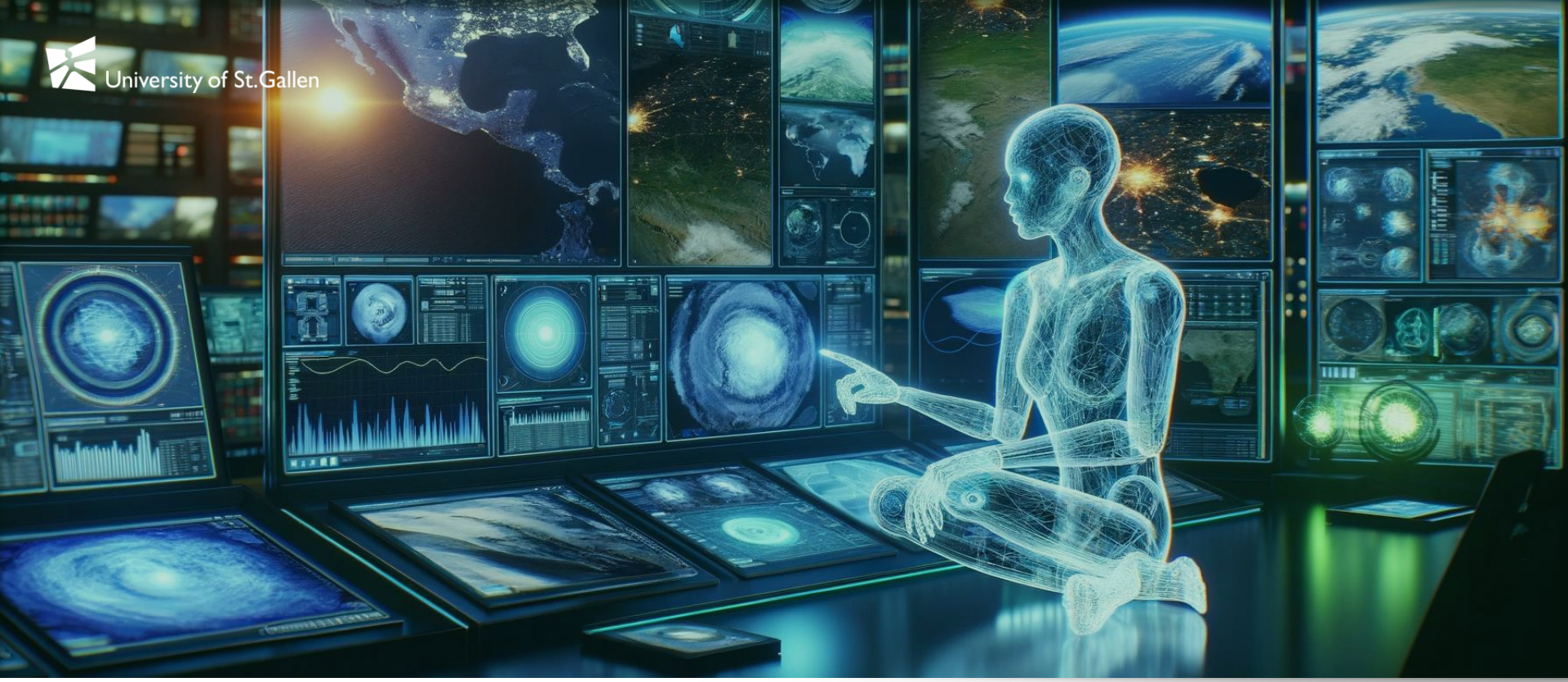
- ✓ ResNet / RNN implementation
- ✓ ViT Implementation
- ✓ BERT implementation
- ≈ Self-/Cross Attention

## Fusion

- ✓ Hadamard
- ✓ Concatenation

## Prediction

- ≈ Multi-Task Setup




# Any questions?

# References

# References

Lobry, S., Marcos, D., Murray, J., & Tuia, D. (2020). RSVQA: Visual Question Answering for Remote Sensing Data. IEEE Transactions on Geoscience and Remote Sensing, 58(12), 8555-8566.

# Appendix

# **Training Configuration** | We further increased efficiency by adjusting our training configuration

**Data Loader**

- In the original train loop, the answer and questions pairs were loaded **separately onto the GPU**
- To save time, we load all Q&A pairs in GPU memory **during initialization** to save time

**Detach & CPU**

- Not detached tensors and tensors in GPU **save additional information**, such as gradients etc.
- By detaching and moving the saved embeddings to the CPU before saving, we save **memory on GPU & disk**

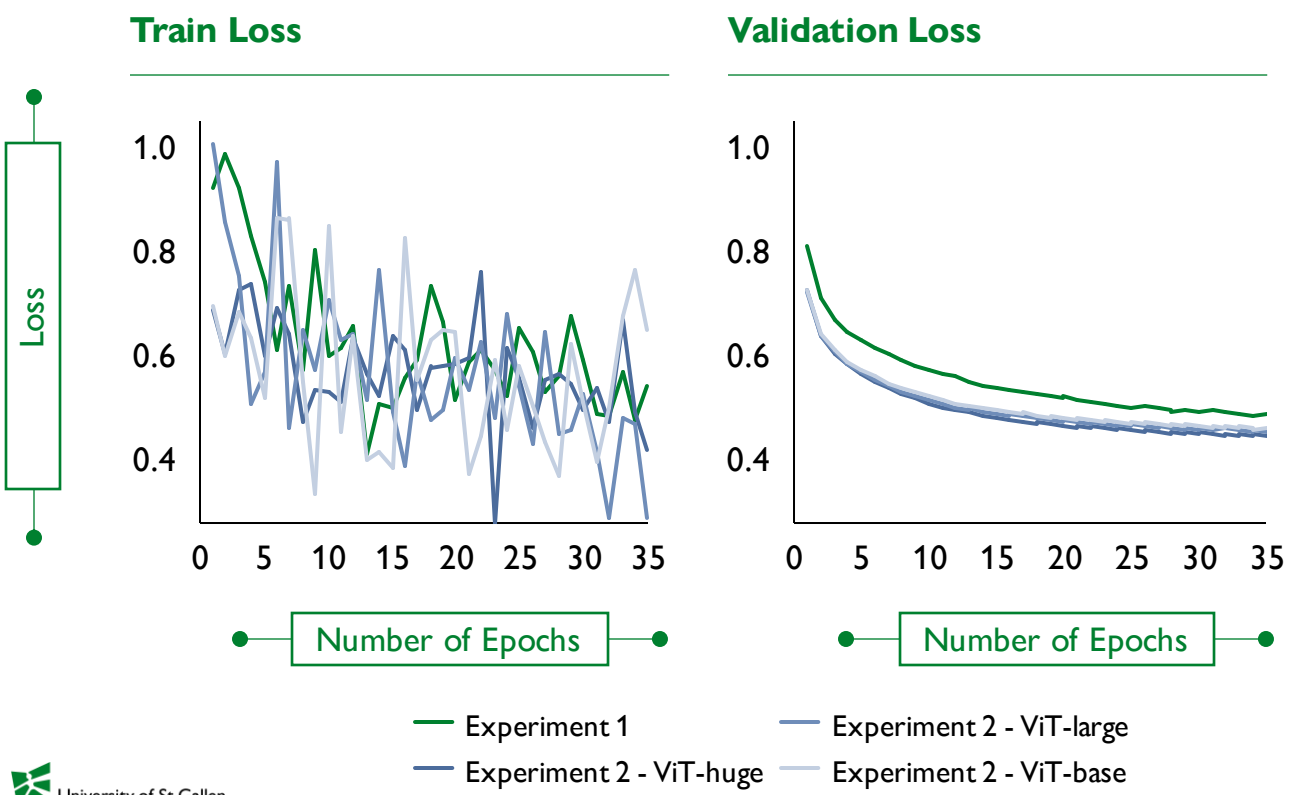
**Other techniques**

- **Pinned Memory:** Tensors are loaded into pinned memory to improve the transfer btw. CPU and GPU
- **Persistent Workers:** Workers stay alive between data batches to reduce overhead of creating new workers
- …

With these changes, the training run now only takes 4.5h

These adjustments enable us to run many experiments with varying configurations

# **Initial Results** | Train and Val Loss over Epochs




Looking at the validation loss, the experiments also show a very similar trajectory.

All ViT sizes perform better in terms of validation loss and the differences are very minimal

We can also see that after epoch 20 the validation loss only decreases marginally, especially with the ViT