Various AI solutions and applications

By: Sikandar Batra

M.S Chemical Engineering, Carnegie Mellon University

Table of Contents

Image Classification and Object Detection

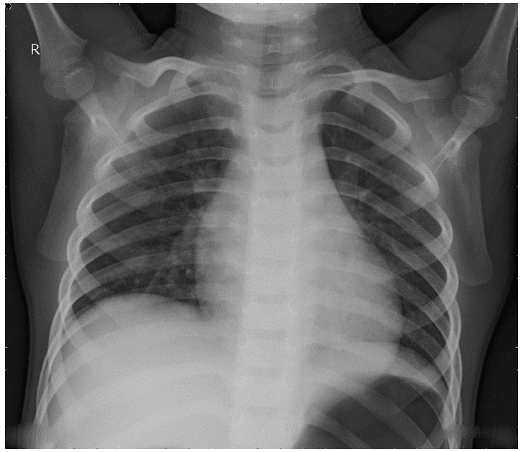
AI-Deep Learning

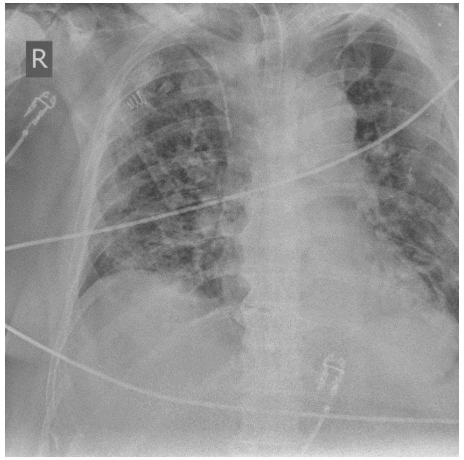
Application of Deep Learning- Neural Networks

- Convolutional Neural Network (CNN): Classification and interpretation of images , audio files etc.
- Recurrent Neural Networks (RNN): Machine Translation, Speech Recognition, Time series Predictions, Speech Synthesis
- Generative Neural Networks: GANS, VAE etc.

Project Application CNN and Generative Network
COVID-19 Detection using Radiography

Dataset





- Contains **Chest X-Ray Images** of 6544 subjects
- **Three classes** - 472 images belong to the COVID-19 class, 4489 belong to the Pneumonia class and 1583 normal lung X-Rays.
- **Imbalanced and Skewed** Dataset

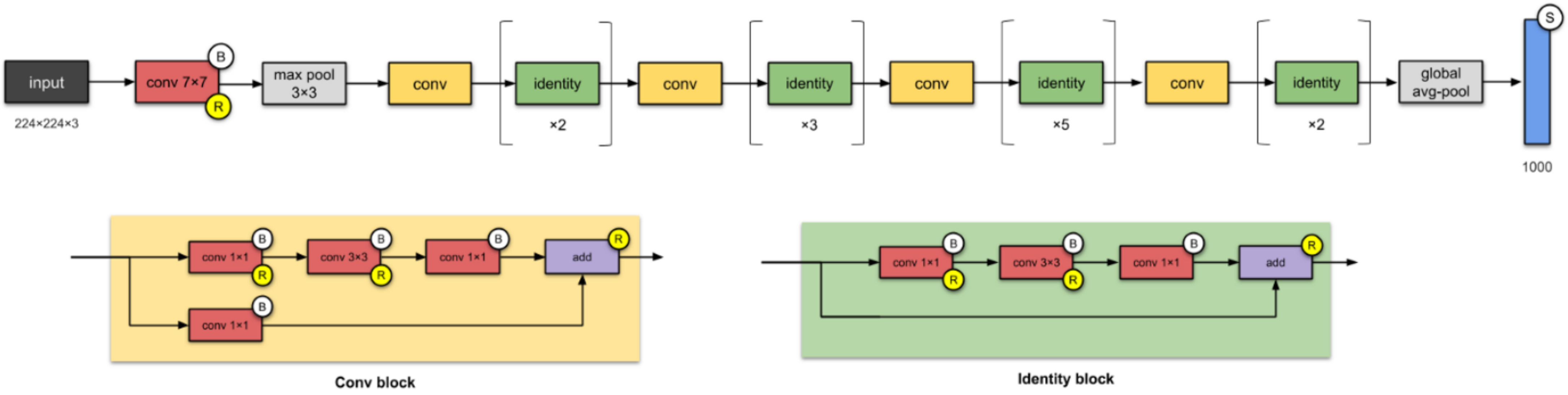
The Approach

- Evaluate CNN-based classifiers to attempt classification of images in **3 Classes: COVID-19, Normal, Pneumonia.**
- If performance is not satisfactory with the CNN models, attempt **Data Augmentation.**
- Train the model on the **Augmented Dataset** with best performing Classifier from the first step.

Classification Networks

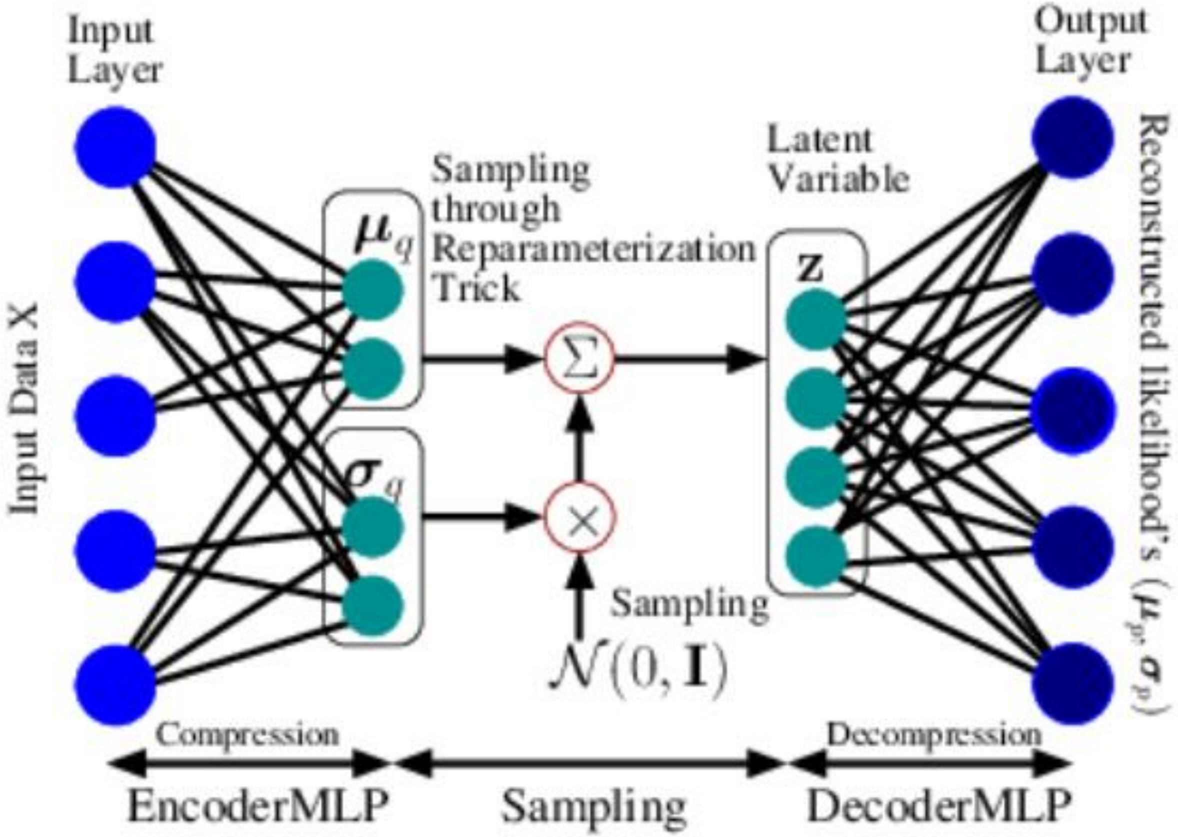
Classification Model Architecture	Classification Accuracy
Vanilla Resnet18	**77.78 %**
Vanilla Resnet34	**88.89 %**
Inverted Bottleneck ResNet	**74.00 %**
VGG-16	**75.00 %**

CNN (ResNet) Classification Network Implemented for the Project

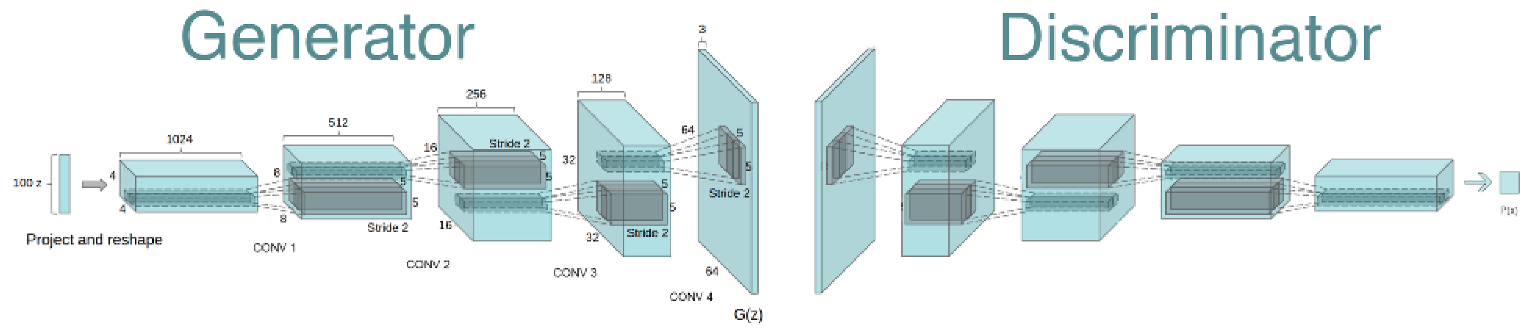


Classification Accuracy of 95% for COVID radiograph images

Variational Autoencoders (VAE)

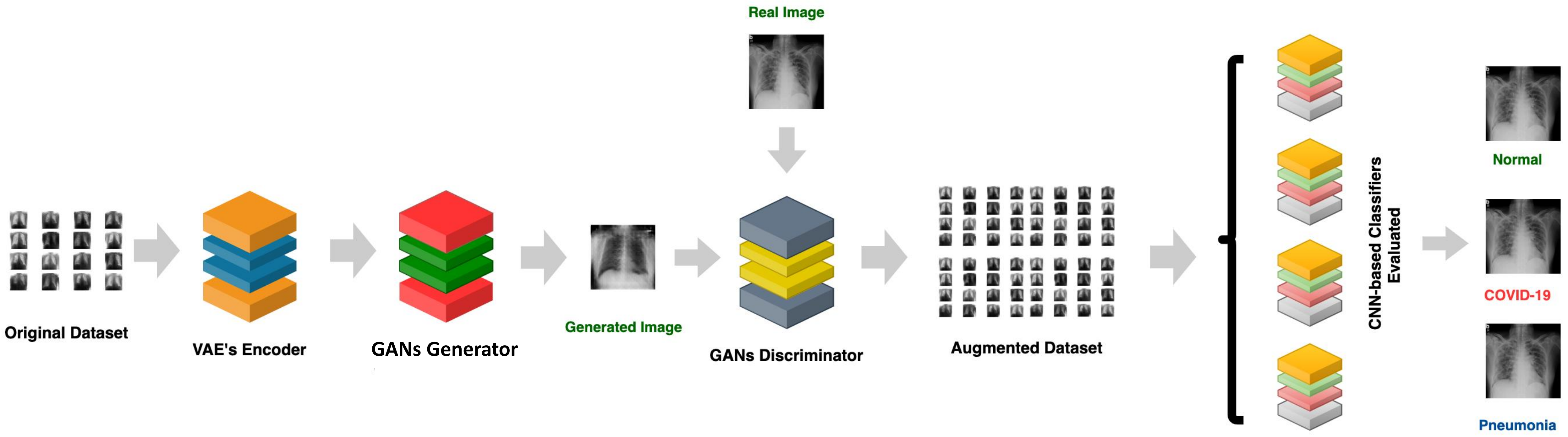


Deep Convolutional GANs - DCGANs



- Transpose Convolutional layers in **Generator**
- Convolutional layers in **Discriminator**
- Latent vector 100 → Image [3, 100, 100] → Real/Fake → Repeat

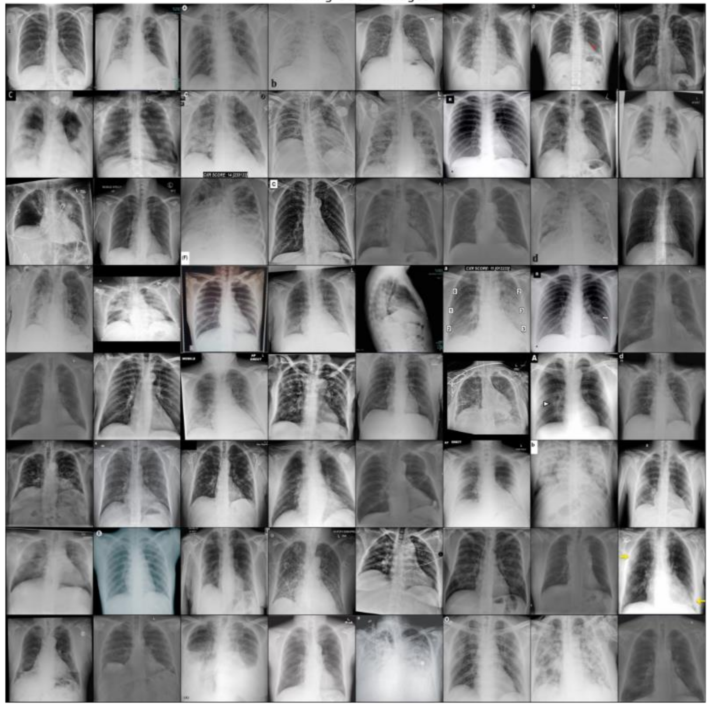
Hybrid VAE-GAN model
Encoder - Decoder - Discriminator



Results of Hybrid Generative Network

Real Images

Training COVID Images



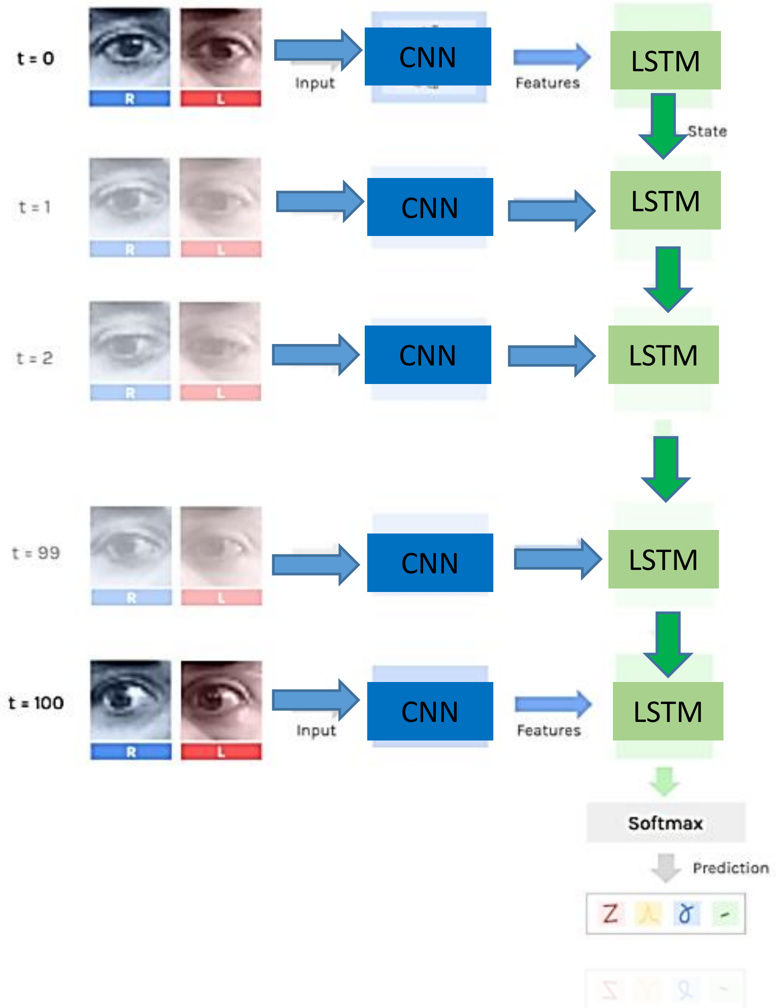
Generated- Normal Images

Generated- COVID19 Images

Classification accuracy of ResNet 34 increased to 95% from data augmentation!!

Project Application RNN and CNN
Real Time Object Detection and Tracking

Network Architecture- Real Time Eye/Pupil detection and tracking



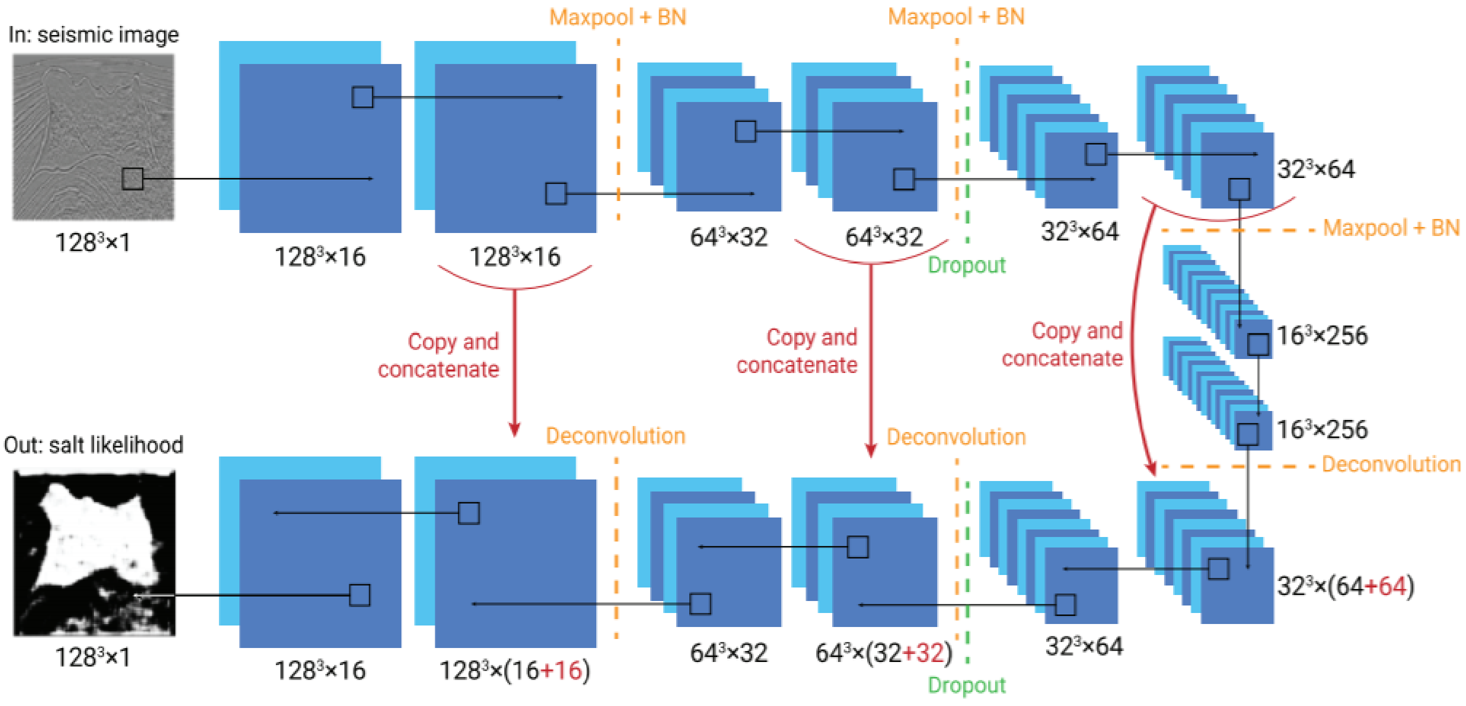
The visual inputs are obtained through a live webcam stream

The CNN extracts visual features from the input, which are processed at each time step by the LSTM (RNN)

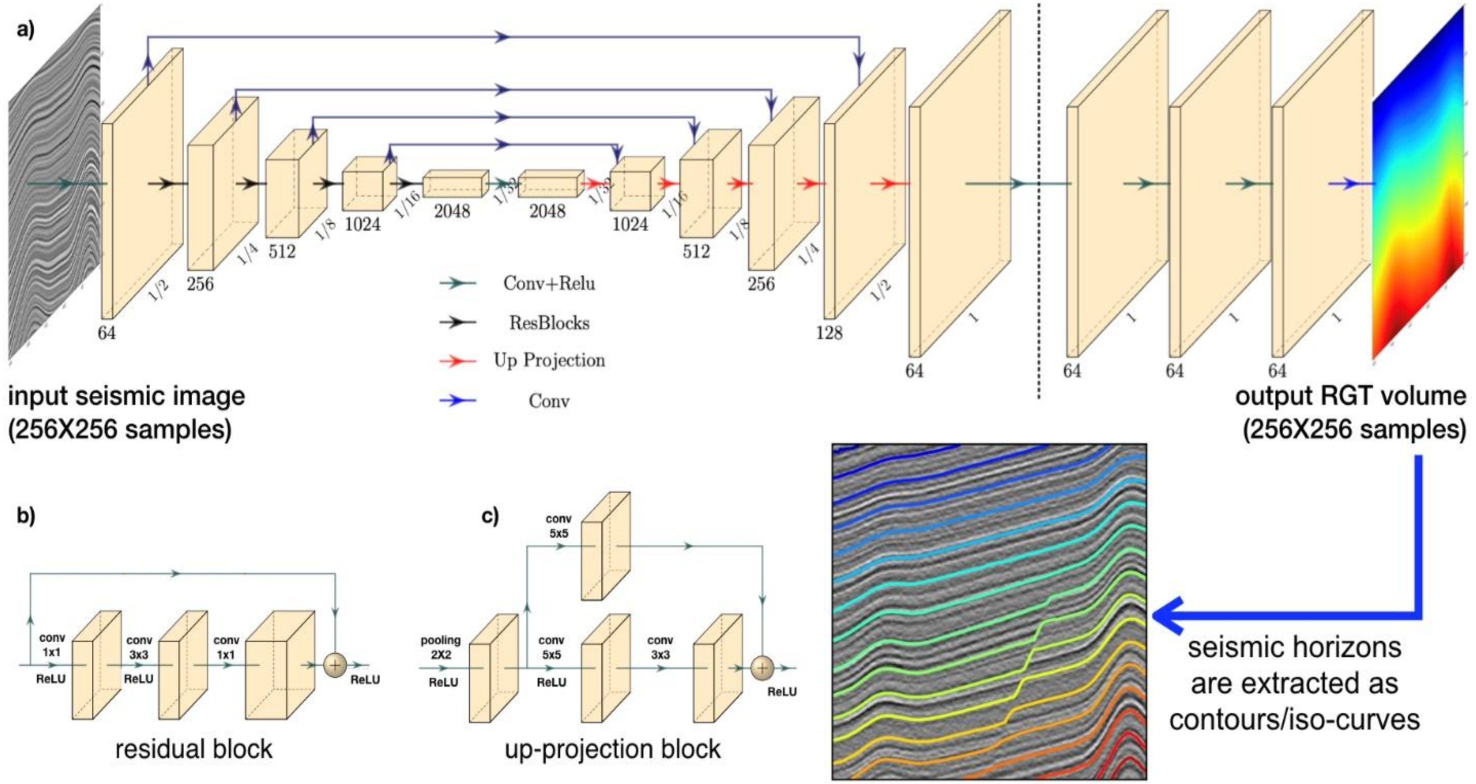
The RNN maps the motion of the eye. Each distinct motion is associated to a specific computer command (clicking, change screen etc.)

Deep Learning Applications
Seismic Studies

3D Salt Segmentation



Constructing Relative Geologic Time (RGT) Images



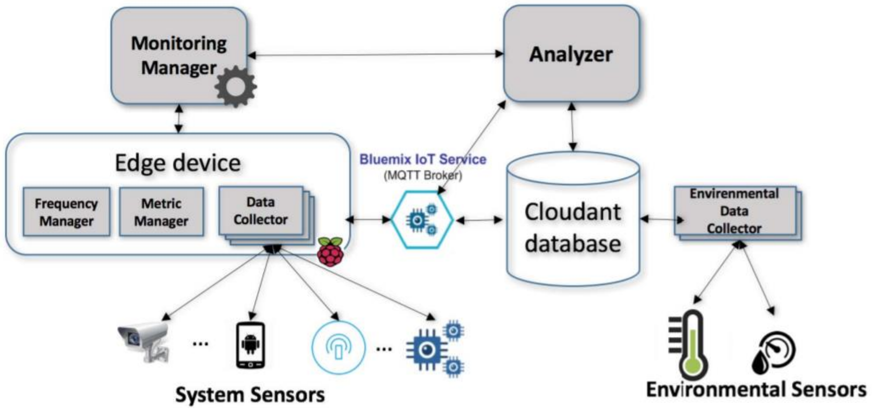
Devising and Optimizing IOT Networks

Project Application- Smart Bus Network

Optimizing Smart Public Bus network (Data from Pittsburg Port Transport Authority)

- Developed a model on GAMS that optimizes the schedule and resource allocation of Pittsburgh public bus lines. Developed model mirrors a multi- layered supply chain network model
- Devised an IOT network responsible for providing real-time parameter values necessary for the GAMS model
- Developed a model on Pyomo and data management applications for on board bus sensors that enables and optimizes adaptive monitoring protocols of said IOT network

IOT Network Optimization




```
while true do
  On event Triggered do
    if the event is environment-related then
      Adapt metrics (add, remove, and retain metrics)
      Compute metric weights
      Optimize metric frequencies
    else
      if the event is metric value-based then
        if there is a need to extend monitoring to correlated metric then
          Adapt metrics (add, remove, and retain metrics)
          Compute metric weights (function of added and retained metrics, and the event)
        else
          Compute metric weights (function of retained metrics and the event)
        end if
        Optimize metric frequencies
      end if
    end if
  End Event
end while
```

Algorithm 1: Metric Adaptation and Frequencies Optimization

$$\max \sum_{i\in I}\sum_{j\in J} w_i * X_{ij} \tag{1}$$

$$\text{s.t.} \sum_{j\in J} B_{ij} = 1, \quad \forall i \in I \tag{2}$$

$$\sum_{k\in K} Interval_{ki} = 1, \quad \forall i \in I \tag{3}$$

$$X_{ij} \geq B_{ij}, \quad \forall i \in I, \quad \forall j \in J \tag{4}$$

$$X_{il} \leq (1 - B_{ij}), \quad \forall l \in \{1,\ldots,j-1\}, \quad \forall i \in I \tag{5}$$

$$\sum_{z\in\{j,\ldots,j+k-1\}} X_{iz} = Interval_{ki}, \quad \forall i \in I, \quad \forall j \in J : (j+k-1) \in J, \quad \forall k \in K \tag{6}$$

$$\sum_{i\in I} ut_{ir} * X_{ij} \leq rcap_{jr}, \quad \forall j \in J, \quad \forall r \in R \tag{7}$$

$$X_{ij} \in \{0,1\}, \quad \forall i \in I, \quad \forall j \in J \tag{8}$$

$$B_{ij} \in \{0,1\}, \quad \forall i \in I, \quad \forall j \in J \tag{9}$$

$$Interval_{ki} \in \{0,1\}, \quad \forall k \in K, \quad \forall i \in I \tag{10}$$

Adaption of Metric Set

- The set of metrics being monitored can be adapted/changed by only two type of events.
- The first type of event is a value related event.
  - A value related event causes the need to retain and extend the monitoring of correlated metrics and eliminates the need to monitor non-correlated metrics.
  - A good example of a value related event in the context of our primary bus route/minimum wait time model is a sudden increase in engine temperature over a predefined threshold.
  - A sudden increase in engine temperature would adapt the set of metrics to include the monitoring of metrics like fuel economy, fuel quality, oil pressure and bus velocity but eliminate or lessen the frequency of monitoring metrics like traffic flow or number of people getting on to the bus at each stop.
- The second type of event that can trigger the metric set to adapt is an environment-related event.
  - Every environment related event is designated with an adaption rule that stipulates what metrics should be added, retained and removed when said event occurs.
  - A pertinent example of an environment related event in the context of our primary bus route/minimum wait time model is the occurrence of rain.
  - The occurrence of rain would add metrics like traffic flow, length of route, bus velocity and number of people getting on the bus at each stand to the metric set but eliminate metrics like engine temperature, oil pressure and tire pressure.

Computation of Metric Weight

- Each metric added or retained to the metrics is given an associative weight. The associative weight of the metric will facilitate the algorithms ability to optimize the frequency each relevant metric should be measured at.
- The methodology to compute the associative weight of the pertinent metrics is dependent on the type of event which triggered the metric set to adapt.
- If the type of the event is a value related event and the event extends the need to add multiple new correlated metrics to the set, the weight of the newly added correlated metrics is computed as an aggregation function of the weights of the metrics that triggered the event, and a proportionate value (dependent on correlation coefficients between new and retained metrics) of the weights of the of metrics that are correlated with the newly added metrics.
- If the value related event does not extend the need to add new metrics to the sets, the new weights of the metrics that triggered said event are computed with a non-decreasing function of the absolute difference between the monitored values of said metrics and the predefined threshold values the monitored values surpassed.

Computation of Metric Weight-Cont.

- If the type of trigger event is environment based, the weights of the newly added metrics to the set are computed by:
  - An aggregation function of the correlation coefficient relating the new metrics with the environment-related event, and a proportionate value (dependent on correlation coefficients between new and retained metrics)) of the weights of the of metrics that are correlated with the newly added metrics.

Frequency Optimization Model

- The following are a list of sets/parameters of a model that maximizes the monitoring frequency of the metrics with the higher weights:
  - I – set of metrics to be monitored
  - K- set of possible monitoring frequency time intervals (eg. monitor every 10 ms) of metrics ordered from lowest to highest interval value. The difference between two successive elements of K is equal to the minimum interval value of K.
  - J- set of possible time stamps during which monitoring can occur. The elements in J are ordered and step size is equal to a pre-defined minimum precision value
  - R- set of resources used to monitor respective metrics
  - $Rcap_{r,j}$ – the capacity of a resource at time stamp j
  - ut_{ir} – the amount of resources, r utilized by metric i
  - The following is a list of variables part of the optimization model:
  - w_i – weight of metric i
  - $Interval_{ki}$ –A binary variable that is equal to one if metric i is monitored at frequency k and zero otherwise
  - B_{ij} –A binary variable that is equal one if monitoring of metric i starts at timestamp j
  - X_{ij} – A binary variable is equal to one if metric i is being monitored at timestamp j

Frequency Optimization Model-Cont.

- The objective of the model (line 1) is to maximize the total monitoring frequency of all the metrics. The monitoring frequency of every metric is directly proportion to the weight of the metric.
- The first constraint (line 2) guarantees that the monitoring of any metric can only begin once during the time horizon.
- The second constraint (line 3) ensures that every metric can only be assigned only one monitoring frequency time interval.
- The third and fourth constraint outlines the relationship between binary variable X_{ij} and B_{ij}. The relationship is that X_{ij} is 0 at time stamps before and until the time stamp where B_{ij} is equal to one. Post said time stamp, X_{ij} can equal to one.
- The fifth constraint ensures that once the monitoring of a metric starts, the metrics successive monitoring is equal to its pre-allocated monitoring frequency time interval.
- The sixth constraint ensures the resources utilized at a specific time stamp does not exceed the resources available at that very time stamp. The successive constraint define the models binary variables.

$$\max \sum_{i\in I}\sum_{j\in J} w_i * X_{ij} \tag{1}$$

$$\text{s.t.} \sum_{j\in J} B_{ij} = 1, \quad \forall i \in I \tag{2}$$

$$\sum_{k\in K} Interval_{ki} = 1, \quad \forall i \in I \tag{3}$$

$$X_{ij} \geq B_{ij}, \quad \forall i \in I, \quad \forall j \in J \tag{4}$$

$$X_{il} \leq (1 - B_{ij}), \quad \forall l \in \{1, \ldots, j-1\}, \quad \forall i \in I \tag{5}$$

$$\sum_{z\in\{j,\ldots,j+k-1\}} X_{iz} = Interval_{ki}, \quad \forall i \in I, \quad \forall j \in J : (j+k-1) \in J, \quad \forall k \in K \tag{6}$$

$$\sum_{i\in I} ut_{ir} * X_{ij} \leq rcap_{jr}, \quad \forall j \in J, \quad \forall r \in R \tag{7}$$

$$X_{ij} \in \{0,1\}, \quad \forall i \in I, \quad \forall j \in J \tag{8}$$

$$B_{ij} \in \{0,1\}, \quad \forall i \in I, \quad \forall j \in J \tag{9}$$

$$Interval_{ki} \in \{0,1\}, \quad \forall k \in K, \quad \forall i \in I \tag{10}$$