A.L.V.I.N.N

Autonomous Learning Vehicle Integrating Neural Networks

Meet the Team

Josh Bertram, Rockwell Collins - Client

Drs. Jones & Zambreno		- Project Advisors
Bijan Choobineh	(CprE)	- Team Leader
Darren Davis	(CprE)	- Team Leader
Tracy La Van	(CprE)	- Communication Leader
Jesse Luedtke	(CprE)	- Key Concepts Holder
David Schott	(SE)	- Key Concepts Holder
Robert Stemig	(CprE)	- Webmaster

ALVINN

Overview

- Introduction to A.L.V.I.N.N
- Design Requirements
- Technical Overview
- Neural Network Design
- Testing & Results
- Concluding Remarks

http://groups.inf.ed.ac.uk/calvin/youtube-objectsv2/Images/aeroplane_00005989.jpg

Objective

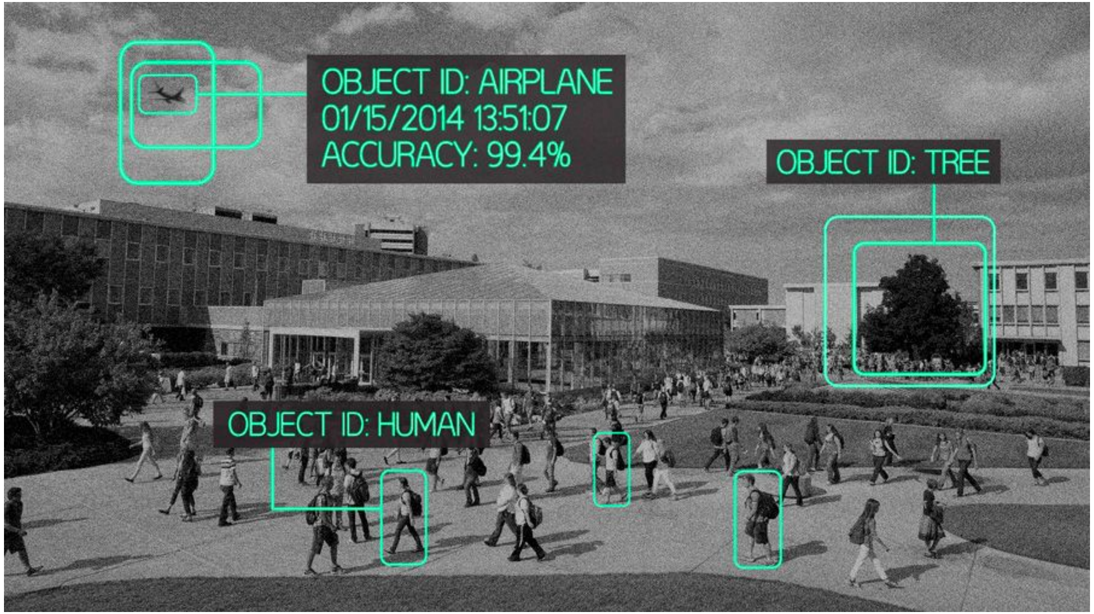
Develop a system for detecting airborne aircraft in collaboration with Rockwell Collins using

- Computer Vision
- Machine Learning
- Neural Networks

http://www.uco.es/~in1majim/proyectos/libpabod/2008_000197_tagged.jpg

Motivation

- Military Applications
  - Survey area for objects of strategic interest
- Civic Applications
  - Find hotspots/survivors during natural disasters
- Research Applications
  - Intersection of computer vision and machine learning



https://cdn-images-1.medium.com/max/1920/1*LgaStRUic1JjYfhdYplClg.jpeg

Functional Requirements

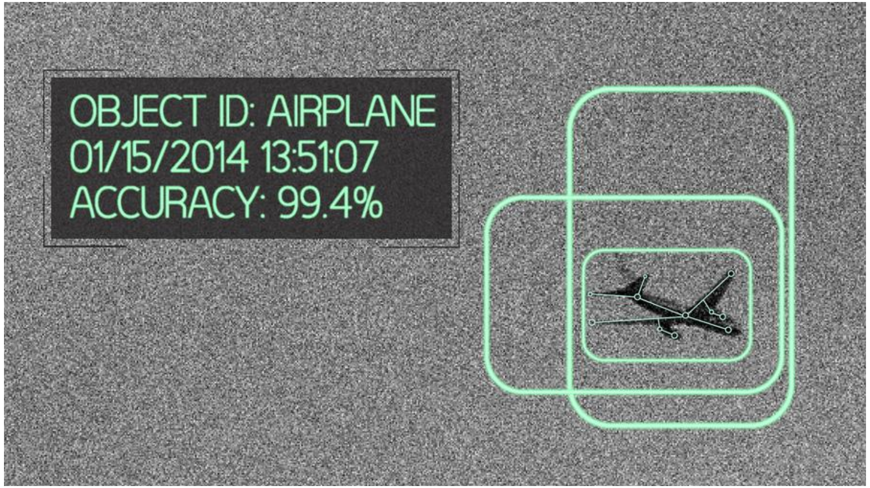
- System must be able to process single image/string of images, and continuous video stream
- Must be able to detect stationary and objects in motion apart from background (and other objects)
- Detect multiple objects of the same or different type
- The image processor must be able to report confidence levels of any identified objects.

http://www.flyhaa.com/content/uploads/2015/09/airplane-courses-reward.jpg

https://www.youtube.com/watch?v=MpPSPQq7oas

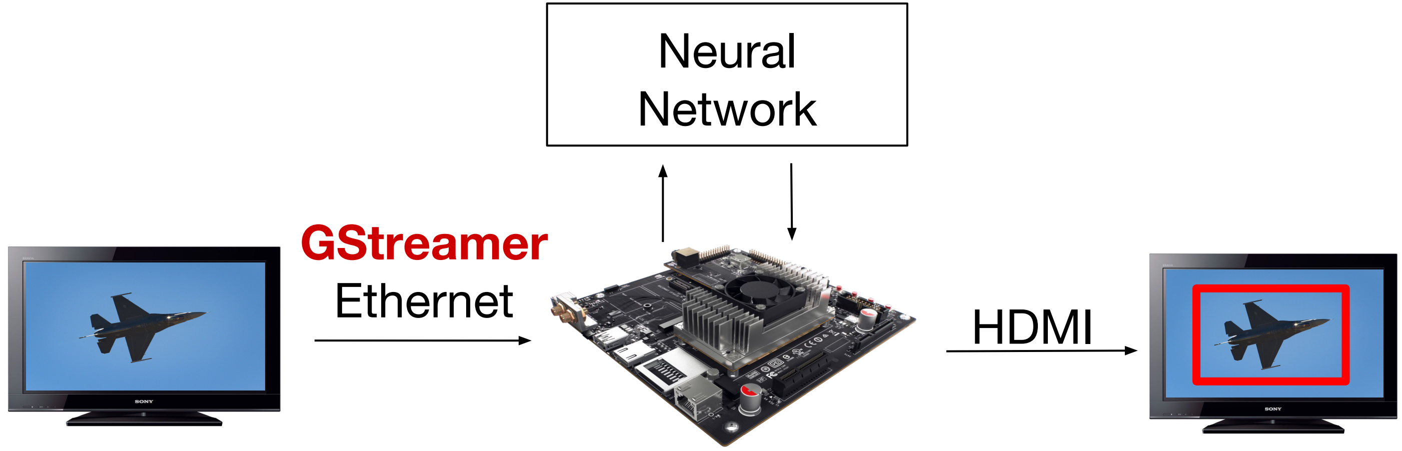
Non-Functional Requirements

- Time Performance
- Extensibility
- Object Detection Accuracy
- Reliability



https://c.slashgear.com/wp-content/uploads/2014/01/object_recognition_plane.jpg

Object Recognition Pipeline

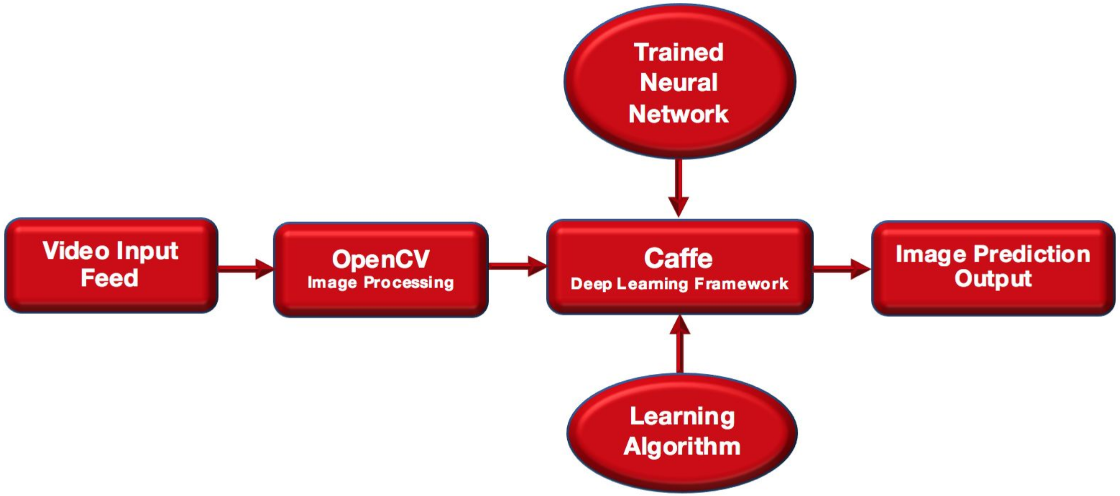


FlightGear Flight Simulator
Host Computer

NVIDIA Jetson TX1

Processed Image
Display Monitor

System Block Diagram



Software

- FlightGear Flight Simulator
  - Multi-platform open source flight simulator
  - Contains wide selection of aircraft
- GStreamer
  - Open source multimedia streaming application framework
  - Streams host computer’s desktop to the embedded board
- OpenCV
  - Open source computer vision library
  - Used to manipulate frames from feed and display result
- Caffe
  - Open source deep learning framework
  - Used for training and execution of neural networks

http://i64.tinypic.com/2dryp3s.jpg

Hardware

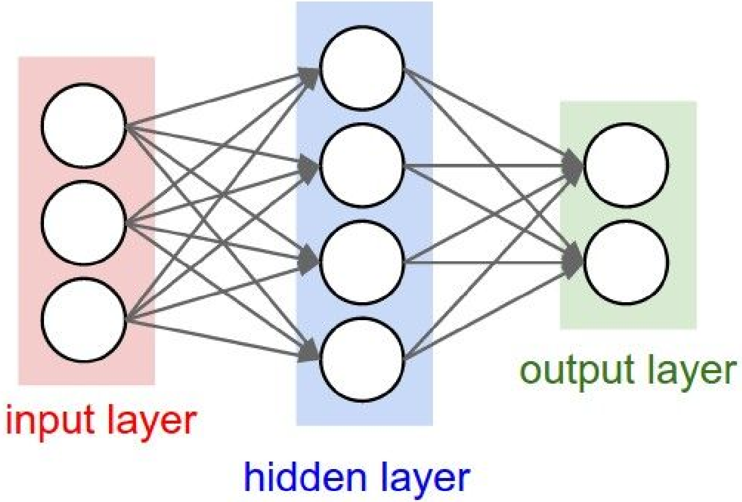
NVIDIA Jetson TX1

- NVIDIA Maxwell GPU
- Linux operating environment (Ubuntu 16.04)
- Supports OpenCV libraries and Caffe framework

https://images10.newegg.com/ProductImage/13-190-006-09.jpg

Artificial Neural Networks (ANN)

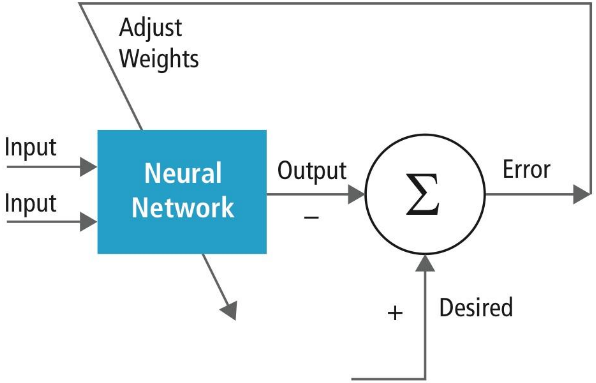
- Machine learning algorithm
- Collection of many neural units (neurons)
- Learns from examples



https://cdn-images-1.medium.com/max/1600/1*BMdnuQUqGEb_FSEmYIxw5g.jpeg

Training

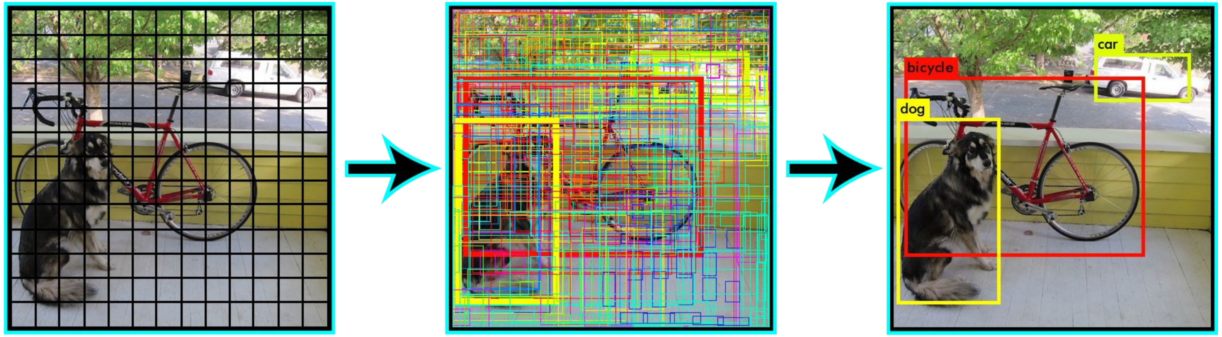
- Model based on BVLC_Caffenet_Model
- ImageNet dataset of airborne objects
  - UAVs
  - Helicopters
  - Airplanes
- ≈ 8,000 training images from ImageNet
- Training performed on different devices
  - Lenovo G50-45 Personal Laptop
  - ISU's High-Performance Computing Node (GPU)
- Achieved a classification accuracy of 90.9%



https://www.embedded-vision.com/sites/default/files/technical-articles/CadenceCNN/Figure2.jpg

Object Detection with Neural Networks (YOLOv2)

- Partition image into a nxn grid
- Each grid cell reports bounding boxes and associated confidence levels
  - Reflects how confident the model is that a bounding box encloses an object
  - Doesn't say anything about possible object classifications
- Each grid cell also predicts a class
  - Reports a probability P(class|object) for all possible classes ← picks highest class probability
- Confidence level * class prediction yields class-specific confidence level
  - Captures both probability of that class appearing in the box and how well predicted box fits object



https://pjreddie.com/media/image/model2.png

Testing - Images

Performance Statistics (%)

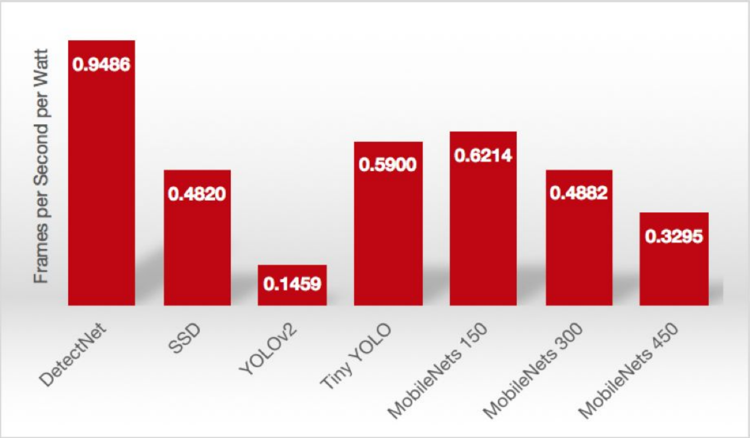
Model	Sensitivity	Precision	Negative Predictor Value	Accuracy	Miss Rate	Fallout
DetectNet	76.92	88.24	52.63	75.47	23.08	28.57
SSD	56.41	100.00	45.16	67.92	43.59	0.00
YOLOv2	71.79	100.00	56.00	79.25	28.21	0.00
Tiny YOLO	46.15	100.00	40.00	60.38	53.85	0.00
MobileNets 150	25.64	100.00	32.56	45.28	74.36	0.00
MobileNets 300	66.67	100.00	51.85	75.47	33.33	0.00
MobileNets 450	71.79	100.00	56.00	79.25	28.21	0.00

Testing - Video

Performance Metrics

Model	FPS	Memory Usage (GB)	Power (mW)
DetectNet	9.60	2.34	10120
SSD	5.57	2.57	11555
YOLOv2	1.75	3.23	11994
Tiny YOLO	6.20	2.21	10508
MobileNets 150	4.15	2.73	6678
MobileNets 300	3.56	2.84	7292
MobileNets 450	3.02	3.04	9166
Idle CPU	n/a	1.16	3343

Image Throughput per Watt



Results

- DetectNet
  - (+) Highest sensitivity and throughput
  - (-) Highest fallout rate
- YOLOv2
  - (+) Highest accuracy (tied with GM450)
  - (-) Highest memory usage and power consumption
- Tiny YOLO
  - (+) Good FPS performance
  - (-) Lowest sensitivity
- SSD
  - (-) Overall low performance
- Google MobileNets (150/300/450)
  - (+) 150: Lowest power consumption
    (-) Lowest performance statistics
  - (+) 450: Highest accuracy
    (-) Subpar throughput
  - (+) 300: Good overall option

Challenges & Mitigation

- Background Knowledge
  - Machine learning field is unfamiliar territory to most of us
  - Glean advice from project stakeholders
- Hardware Difficulties
  - NVIDIA Jetson TX1 onboard camera
  - Software setup
- Video Streaming Support
  - Webcam, video file, JPEG stream (GStreamer)
  - GStreamer feeding into proprietary neural networks (DetectNet)
- Communication
  - Team Leads communicate with advisors & client
  - Communication Lead organizes communication within the team

Concluding Remarks

- Learning experience
  - Multidisciplinary project
  - Designed from conception to implementation
- Several areas for improvement
  - Better testing methods
  - More data
  - Retraining

Questions?

From Left: Bijan Choobineh, David Schott, Jesse Luedtke, Tracy La Van, Darren Davis,Robert Stemig.

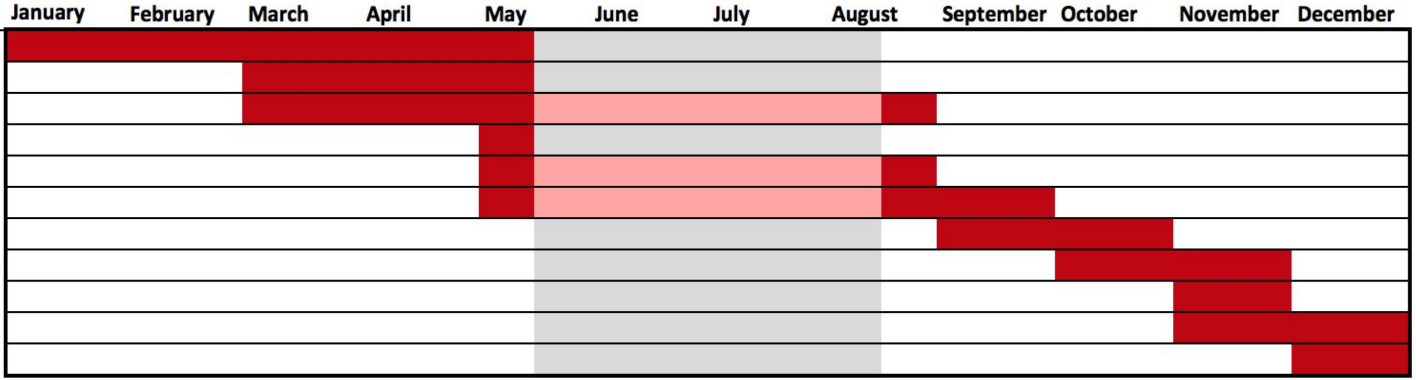
Visit our website!

Performance Statistics

- **Sensitivity** (true positive rate) was measured to see how each network could correctly identify an aircraft in an image when one was present
- **Specificity** (true negative rate) - used measured to see if a network could correctly identify that no aircraft was present in an image when there was no aircraft in the images
- **Precision** - used to see the percent of aircraft being correctly classified from the total number of aircraft the network detected
- **Negative predictive value** - used to see what percent of objects correctly classified as not an aircraft from the total number of objects the network not classified as not an aircraft
- **Accuracy** - used to analyze the total percent of classifications correct
- **Miss rate** (false negative rate) - used to see the percent of time the network didn't think an object was an aircraft but it really was an aircraft
- **Fallout** (false positive rate, or “false alarm”) - used to see the percent of time the network thought the object was an aircraft when it was not an aircraft.

Project Plan

ID	Task Name
1	Phase I: Education
2	Software/Hardware
3	Phase II: Detect Simple 'X'
4	491 Demo
5	Software/Hardware Setup
6	Phase III: Detect Other Objects
7	Phase IV: Classify Simple Objects
8	Phase V: Classify Complex Objects
9	Phase VI: Real-Time System
10	(Possible Bonus Features)
11	Final Demo



Computer Vision

- Interdisciplinary field exploring computational ways to extract information from digital images or videos
- Sub-domains include:
  - Image restoration
  - Video motion detection
  - Object recognition e.g. identification of a vehicle, OCR, etc.