KI-1-AI-Security

Date: 11/07/23

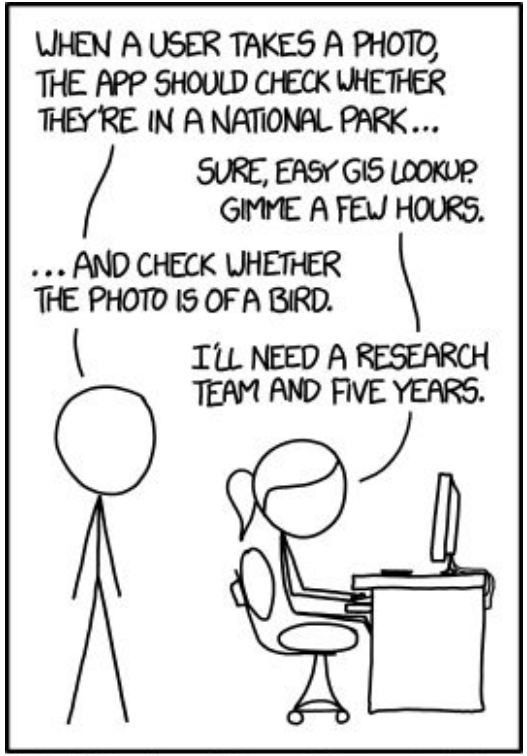
Senior Project 4850 - 02

Fall semester

Sharon Perry

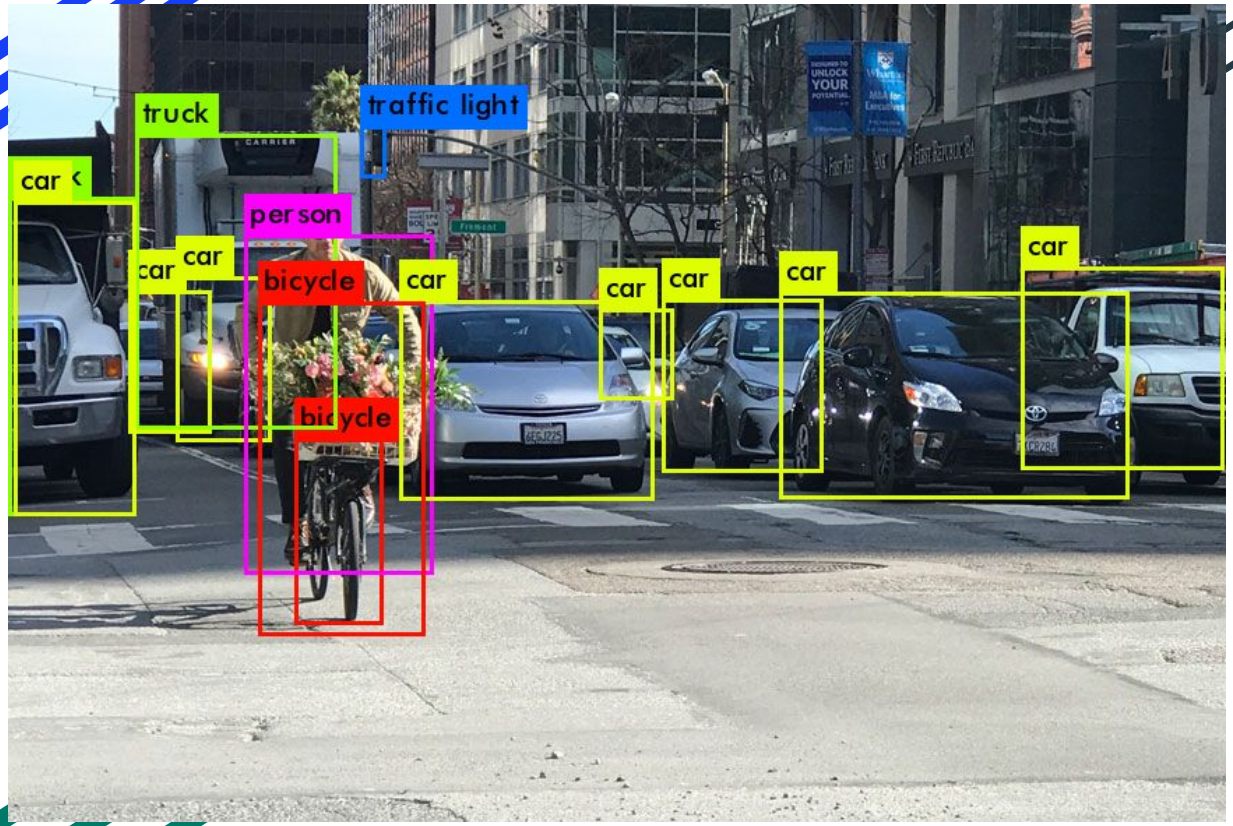
By: Matthew Krupczak, Zeke Lipscomb, Babrah Koroma, Kyle Bratcher

For Dr. Kazi Aminul Islam

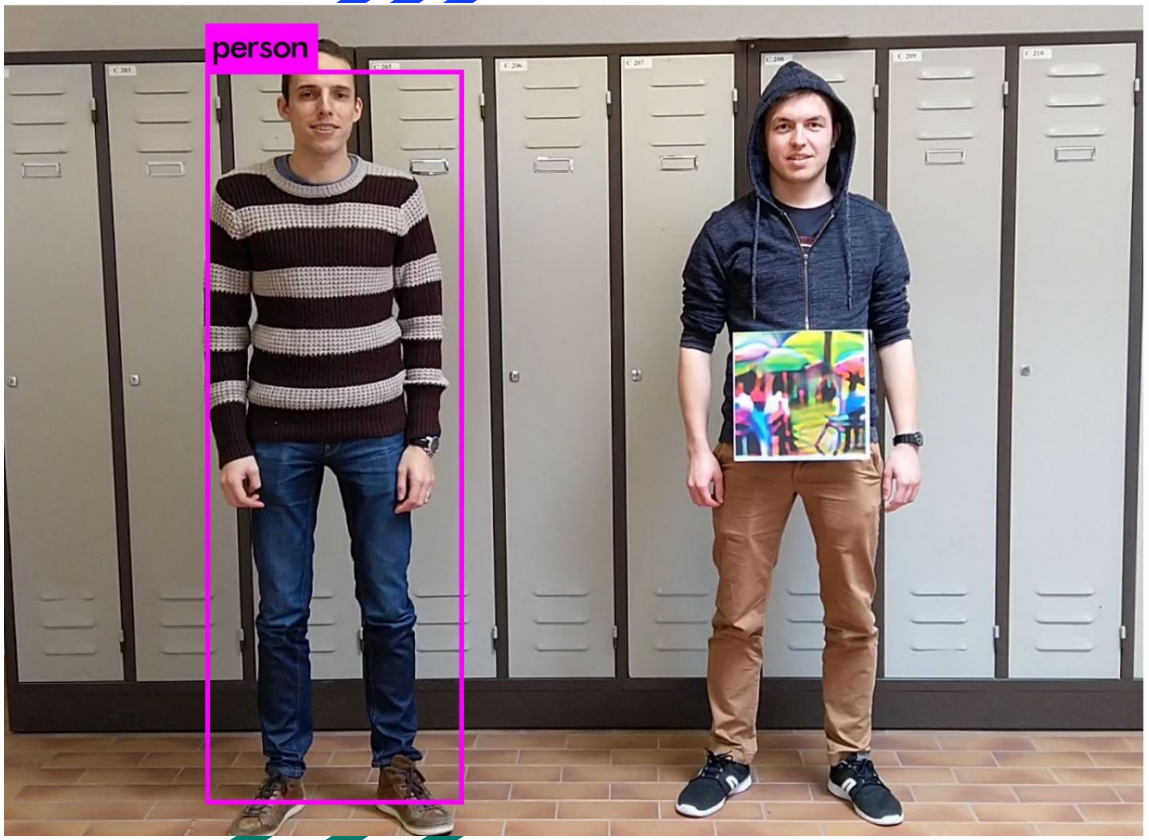


IN CS, IT CAN BE HARD TO EXPLAIN THE DIFFERENCE BETWEEN THE EASY AND THE VIRTUALLY IMPOSSIBLE.

xkcd.com/1425 September 2014



AI Object Detection is a miracle!



we created this^

An adversarial pattern disrupts object detection (Thys et al. 2019)

What if this had been something ***really important?***

Zeke - Team Lead

Project Name:	KI-1-AI-Security														
Report Date:	9/7/2023														
					Milestone #1				*Milestone #2*						*Milestone #3*
Deliverable	**Tasks**	**Complete%**	**Current Status Memo**	**Assigned To**	**09/02**	**09/09**	**09/16**	**09/23**	**09/30**	**10/07**	**10/14**	**10/21**	**10/28**	**11/04**	**11/11**
Plane Obj Detection Model	Obtain access to KSU HPC	100%	Completed	Kyle, Matthew	2	2									
	YOLOv8 model jupyter notebook	50%	Technique is ready	Kyle	1	4									
	YOLOv7 model manual training	10%	Torque scripts available	Matthew		2									
	Model testing and finalization	0%		All			7	10							
Train Adversarial Model	Fork codebase to team GitHub	0%		Kyle, Matthew				1	1						
	Planning for newer YOLO port	0%		Kyle, Matthew, Sumaiya				10	10	2					
	Porting to newer YOLO	0%		Kyle, Matthew, Sumaiya						8	10	10	10		
	Testing and Debugging	0%		All									8	8	
	Prepare training adversarial model	0%		Kyle, Matthew, Sumaiya										8	
Generate Defeat Pattern	Obtain output from trained model	0%		Matthew										2	4
	Validation testing (mAP)	0%		Babrah											8
Code and Docs	Code Cleanup	0%		Kyle, Matthew, Sumaiya											
	Documentation and reproducabilit	0%		All											
Final report	Presentation preparation	0%		Zeke											
	Poster preparation	0%		Babrah , Kyle, Matthew											
	Final report submission to D2L and project owner	0%		All											
			Total work hours	**198**	**3**	**8**	**7**	**21**	**11**	**10**	**10**	**10**	**18**	**18**	**12**

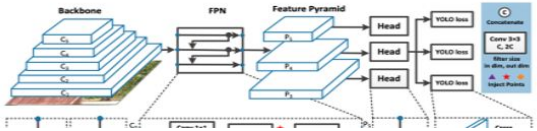
RESEARCH PROJECTS

Design Document

With the advancement of Artificial Intelligence (AI) technology, it is used in every part of our life, e.g., finance, security system, autonomous vehicle, cybersecurity, and transportation system. But attackers can also manipulate this AI technology with specially crafted input. The Adversarial example is one of the earliest adversarial attacks in AI models where an attacker can craft a poison sample during the inference phase. Usually, the object of the attacker is to hide objects from AI- based detectors. This might create a catastrophic effect on many security-critical systems. Government organizations, such as US Navy, US Army, and many private sectors, e.g., Uber and Google, use remote sensing data. Satellite imagery, one of the most popular remote sensing products, is widely used to monitor objects remotely. In this project, Students will explore the potential vulnerabilities of the AI models in remote sensing-based AI models and developing mitigation techniques.

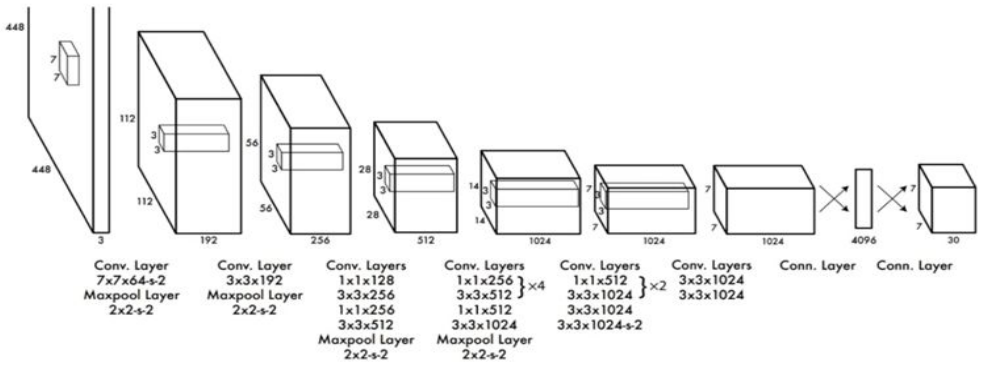
The team will use YOLOv8 to explore and exploit the potential vulnerabilities in the recognition AI. The YOLO (You Only Look Once) v8 model is the latest in the family of YOLO models. YOLO models are single stage object detectors. In a YOLO model, image frames are featurized through a backbone. These features are combined and mixed in the neck, and then they are passed along to the head of the network YOLO predicts the locations and classes of objects around which bounding boxes should be drawn.

"YOLO conducts a post-processing via non-maximum supression (NMS) to arrive at its final prediction." (Solawetz, 22)



How Does YOLO Work

The YOLO algorithm takes an image as input and then uses a simple deep convolutional neural network to detect objects in the image. The architecture of the CNN model that forms the backbone of YOLO is shown below.



The Architecture. Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection.

Babrah Koroma

- Senior major in Computer Science
  - Software engineer and Web developer
- Role : Administrative and Data Science

Documents

- Requirements
- Design
- Website
- Final Project Report

KI-1-AI-Security - Requirements

Date: 09/23/23

4850 - 02

Fall semester

Sharon Perry

Project Team

Roles	Name	Major responsibilities
Project owner	Dr. Kazi Aminul Islam	Provide direction for the project and set goals
Team leader	Zeke Lipscomb	Lead the team and own administrative aspects of the project
Team members	Kyle Bratcher	Individual Contributor for code
	Babrah Koroma	Administrative and Data Science assistance
	Matthew Krupczak	Individual Contributor for code
Research Assistant	Sumaiya Tasneem	Grad student added to project at request of Dr. Islam. Technical assistance
Advisor / Instructor	Sharon Perry	Facilitate project progress; advise on project planning and management.

Kyle

Zeke

Babrah

Matthew

RESEARCH PROJECTS

Requirements

We review and study the tutorial links listed in the content section and assignments

Project requirements and design

Gantt Chart

Project under facility research

Activities

Form teams

All teams members study the research project materials

Distribute tasks to team members.

Matthew and Kyle work mainly on coding

Babrah and Zeke work mainly on documents/ submission

All members obtain access to KSU HPU J-160

Daily meetings on Thursday at 4:15pm to 4:45pm.

All teams Plan Obj Detection Model

Work on YOLOv8 model jupyter notebook

Work on YOLOv7 model manual training

All teams work on model testing and finalization

All team members train Adversarial Model

Fork database on team Github

Planning for newer YOLO port

Porting to newer YOLO

Testing and Debugging the code

Prepare training adversarial model

All generate Generate Defeat Pattern

Obtain output from trained model

Validation testing (mAP)

Code and Docs

All members work on code cleanup

All members work on documentation and reproducibility

Use google doc/Spreadsheet to complete all of the documents

All teams create website to demo research materials using github

ex https://ki-1-ai-sec.github.io

Final report

All members work on Presentation preparation

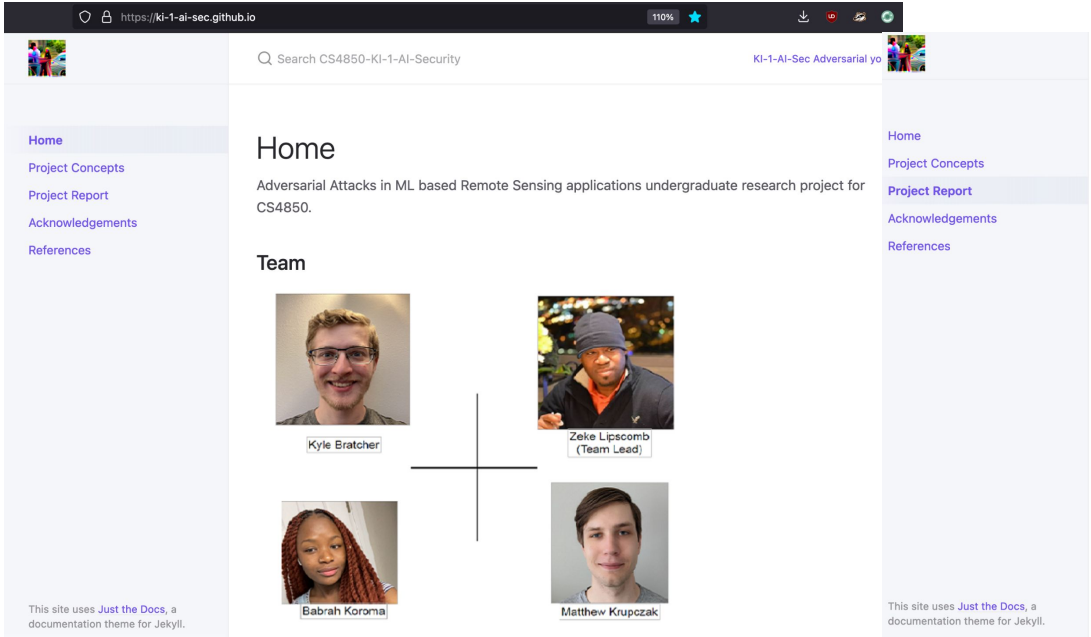
All members work on presentation preparation

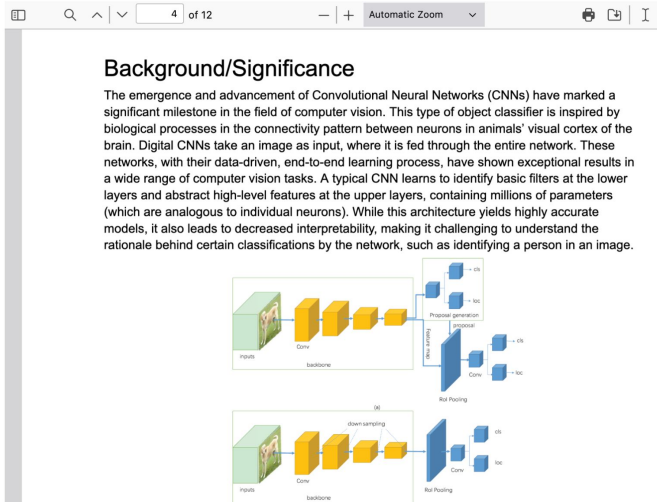
All members work on final report and submission to D2L and project owner

Site Specific references use to complete the project

End of document

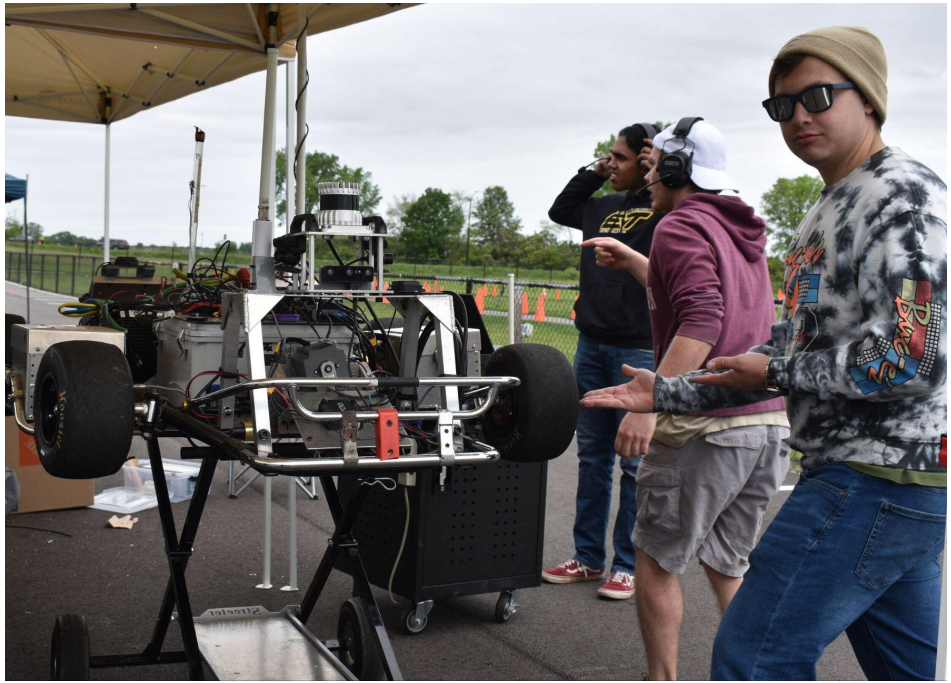
Website using Github Pages





Matthew Krupczak

- Senior class of 2023
- Member of KSU Electric Vehicle Team
- Founder of Theta Informatics LLC

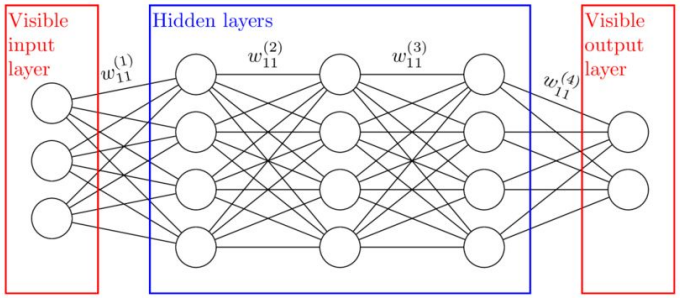


Brandon Solon with ‘Voltron’, KSU EVT’s competition racekart, at AKS 2023

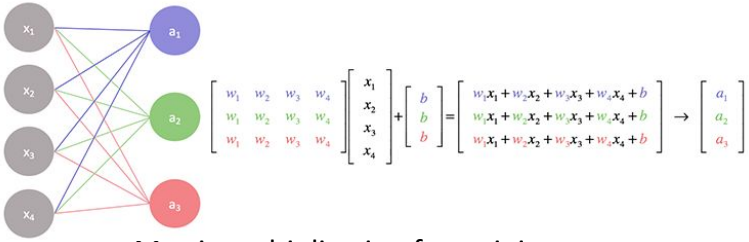
Coneslayer - YOLOv7 object detection model

HPC - High Performance Computer

An NVIDIA Ampere V100S 32 Gb HPC (KSU's HPC has 4 cards per node)

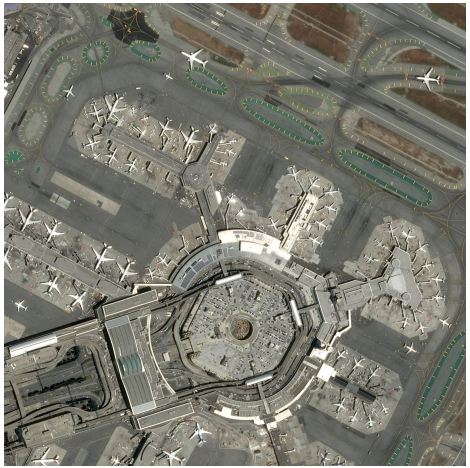


A feed-forward neural network

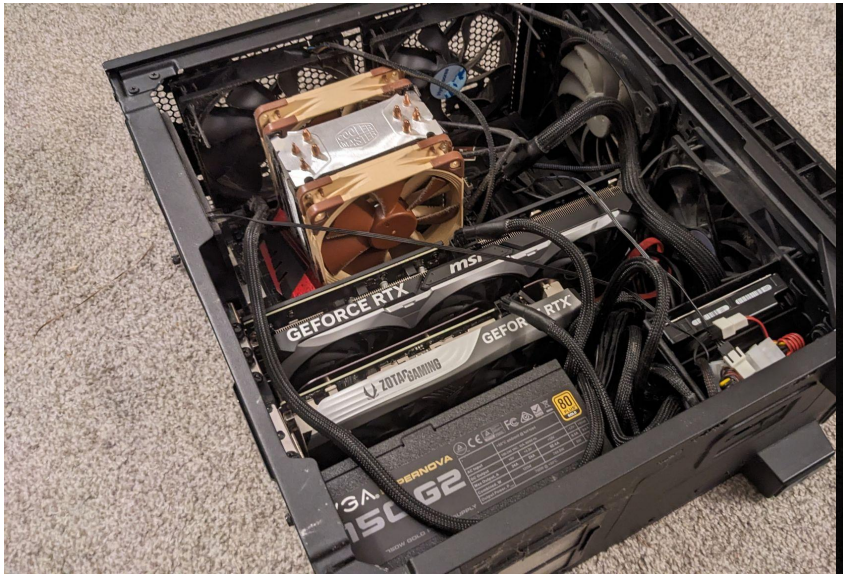


Matrix multiplication for training

Airbus Defense and Space - Airplane detection

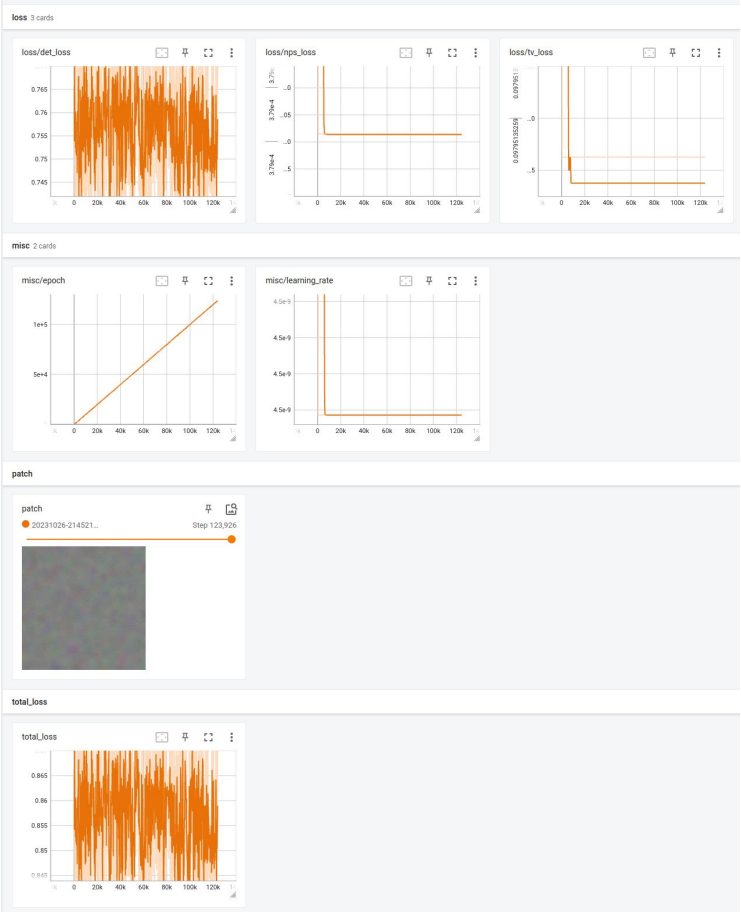


“We have HPC at Home”
HPC at Home:



Initial training runs:

- Used **tensorboard** running locally to visualize training run with **pytorch**
- Increased batch size and learning rate from original experiment
- Faster iteration for early stages of development



Initial Development

IP[y]: Notebook spectrogram Last Checkpoint: a few seconds ago (autosaved)

IPython (Python 3)

File Edit View Insert Cell Kernel Help

Code Cell Toolbar: None

Simple spectral analysis

An illustration of the Discrete Fourier Transform using windowing, to reveal the frequency content of a sound signal.

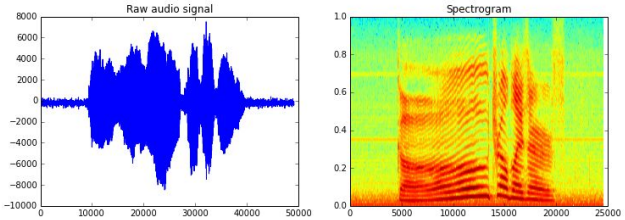
$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \qquad k = 0, \ldots, N-1$$

We begin by loading a datafile using SciPy's audio file support:

In [1]:
```
from scipy.io import wavfile
rate, x = wavfile.read('test_mono.wav')
```

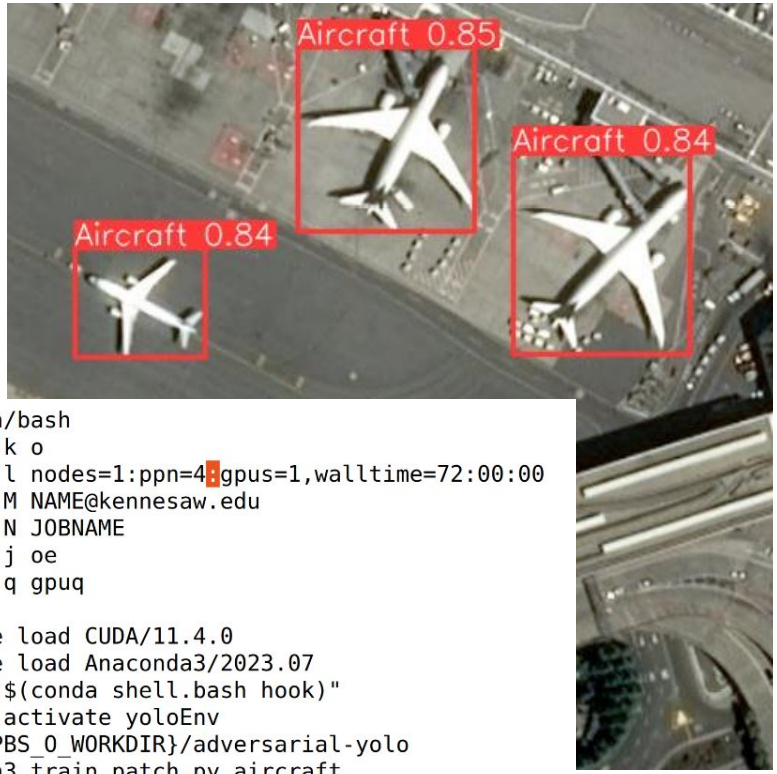
And we can easily view its spectral structure using matplotlib's builtin `specgram` routine:

In [2]:
```
%matplotlib inline
from matplotlib import pyplot as plt
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 4))
ax1.plot(x); ax1.set_title('Raw audio signal')
ax2.specgram(x); ax2.set_title('Spectrogram');
```



Training

20231031-203437_airbus-subset-8	11/2/2023 1:03 PM	File folder
20231030-213350_paper_obj	11/2/2023 12:34 AM	File folder
20231030-213644_paper_obj	11/2/2023 12:34 AM	File folder
20231030-214716_paper_obj	11/2/2023 12:16 AM	File folder
20231029-221146_aircraft	10/30/2023 5:19 PM	File folder
20231029-201627_paper_obj	10/29/2023 5:44 PM	File folder
20231029-032622_paper_obj	10/29/2023 12:17 PM	File folder
20231029-014838_paper_obj	10/28/2023 11:03 PM	File folder
20231029-013649_paper_obj	10/28/2023 9:45 PM	File folder
20231029-010416_aircraft	10/28/2023 9:10 PM	File folder



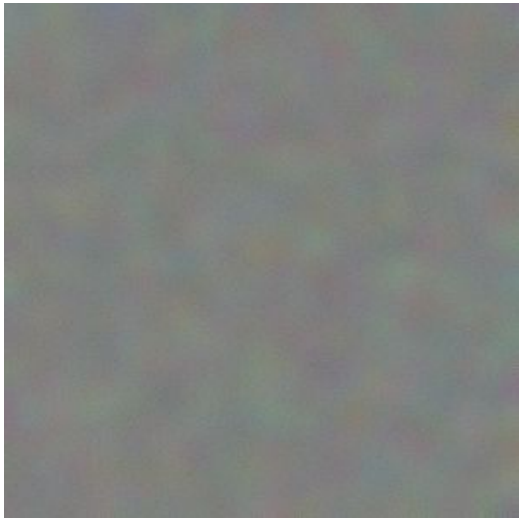

```
#!/bin/bash
#PBS -k o
#PBS -l nodes=1:ppn=4:gpus=1,walltime=72:00:00
#PBS -M NAME@kennesaw.edu
#PBS -N JOBNAME
#PBS -j oe
#PBS -q gpuq

module load CUDA/11.4.0
module load Anaconda3/2023.07
eval "$(conda shell.bash hook)"
conda activate yoloEnv
cd ${PBS_O_WORKDIR}/adversarial-yolo
python3 train_patch.py aircraft
```

Progress?

Box(P	R	mAP50	mAP50-95)
0.954	0.867	0.917	0.689

Box(P	R	mAP50	mAP50-95)
0.922	0.765	0.811	0.557



Questions?