CV Projects with code and dataset

1. Object Detection

Object Detection involves detecting instances of objects in images or videos. It involves identifying and localizing objects within an image and classifying them into pre-defined categories.
Models: MobileNet SSD or YOLO
• Datasets: COCO dataset
• Application domain: Image Processing
• Level: Beginner
• Audience Interest level: Moderate
• Explanation: Object Detection using Region-based CNN
• Codebase:
https://github.com/tensorflow/models/tree/master/research/object_detection
• Research paper: https://arxiv.org/abs/1506.01497

2. Image Segmentation

Image Segmentation involves dividing an image into multiple segments or regions, each of which corresponds to a different object or part of the image. It involves assigning a label or class to each pixel in the image, based on the pixel's characteristics and its relationship to other pixels in the image.
• Models: U-Net, Mask R-CNN, FCN
• Datasets: PASCAL VOC, COCO dataset
• Application domain: Medical Image Processing
• Level: Intermediate
• Audience Interest level: High
• Explanation: Image Segmentation using Convolutional Neural Networks
• Codebase:
https://github.com/divamgupta/image-segmentation-keras
• Research paper: https://arxiv.org/abs/1505.04597

3. Single Shot MultiBox Detector (SSD)

This is a single-shot object detection system that uses a deep convolutional neural network to predict object class scores and bounding box offsets. The system is fast and efficient, making it well-suited for real-time object detection tasks.
• Codebase:
https://github.com/weiliu89/caffe/tree/ssd
• Research paper: https://arxiv.org/abs/1512.02325

4. Pose Estimation

This involves estimating the pose (position and orientation) of an object in a given image. It is typically used to determine the orientation of human bodies, faces or objects, and is widely used in applications such as augmented reality, gaming, and human-computer interaction.
• Models: OpenPose, AlphaPose
• Datasets: COCO Keypoint dataset
• Application domain: Human Pose Analysis
• Level: Intermediate
• Audience Interest level: High
• Explanation: Pose Estimation using Deep Learning
• Codebase: https://github.com/CMU-Perceptual-Computing-Lab/openpose
• Research paper: https://arxiv.org/abs/1812.08008

5. Face Detection

This involves identifying and locating human faces in images or videos. It is a crucial component in many facial recognition systems, and often the first step in processing facial images.
• Models: Single Shot MultiBox Detector (SSD)
• Datasets: WIDER FACE dataset
• Application domain: Face Recognition
• Level: Beginner
• Audience Interest level: Moderate
• Explanation: Face Detection using Deep Learning
• Codebase: https://github.com/opencv/opencv/blob/master/samples/dnn/face_detector
• Research paper: https://arxiv.org/abs/1512.02325

6. Lane Detection

This involves identify the lanes and markings on roads in images and videos, typically used in self-driving car applications to assist the navigation. It involves detecting the boundaries of lanes and classifying them as either driving or non-driving lanes to provide real-time lane guidance.
• Models: Canny Edge Detection, Hough Transform
• Datasets: Udacity Self-Driving Car dataset
• Application domain: Autonomous Driving
• Level: Beginner
• Audience Interest level: Low
• Explanation: Lane Detection using Computer Vision techniques

• Codebase: https://github.com/udacity/CarND-LaneLines-P1
• Research paper: https://ieeexplore.org/abstract/document/7960982

7. Optical Flow

This involves calculating the motion of objects and pixels between consecutive frames in a video. It is used in various applications such as video compression, action recognition, and autonomous driving.
• Models: Farneback, Lucas-Kanade
• Datasets: Middlebury Optical Flow dataset
• Application domain: Motion Analysis
• Level: Intermediate
• Audience Interest level: Low
• Explanation: Optical Flow Estimation using Computer Vision techniques
• Codebase: https://github.com/opencv/opencv/blob/master/samples/python/opt_flow.py
• Research paper: https://ieeexplore.org/abstract/document/731667

8. DeepLab

This is a series of deep convolutional neural networks for semantic image segmentation. The systems use atrous convolutions and conditional random fields to generate high-quality segmentation masks.
• Codebase: https://github.com/tensorflow/

9. Object Tracking

This involves locating and tracking objects over time in a video stream. It is typically accomplished by using computer algorithms to analyze sequential frames of the video and determine the object's position and trajectory in each frame.
Models: SORT (Simple Online and Realtime Tracking).
• Datasets: VOT challenge dataset
• Application Domain: Video Processing
• Level: Intermediate
• Audience Interest level: High
• Explanation: Object Tracking using Kalman Filter and Hungarian Algorithm
• Source code: https://github.com/abewley/sort

• Research paper: https://arxiv.org/abs/1602.00763

10. Deeplab v3+

This is an improved version of the DeepLab semantic image segmentation system. The system uses atrous convolutions and deep supervision to achieve high performance on a variety of benchmark datasets.
• Codebase: https://github.com/tensorflow/models/tree/master/research/deeplab
• Research paper: https://arxiv.org/abs/1802.02611

11. Image Classification

This involves assigning a label to an input image, based on its visual content. It involves training a machine learning model on a large dataset of labeled images and using it to predict the class of new, unseen images.
• Models: ResNet, InceptionNet, VGGNet
• Datasets: ImageNet dataset
• Application domain: Image Processing
• Level: Beginner to Intermediate
• Audience Interest level: Moderate
• Explanation: Image Classification using Deep Convolutional Neural Networks
• Source code: https://github.com/pytorch/examples/tree/master/imagenet
• Research Paper: https://arxiv.org/pdf/1512.03385.pdf

12. Car License Plate Detection

This involves locating and extracting the license plate information from a given car image. The extracted information can then be used for various purposes such as vehicle identification and tracking.
• Models: YOLO, Faster R-CNN, Single Shot MultiBox Detector (SSD)
• Datasets: ALPR (Automatic License Plate Recognition) dataset, The Belgian Traffic Sign Recognition database
• Application domain: Image Processing, Transportation
• Level: Intermediate
• Audience Interest level: Moderate
• Explanation: Car License Plate Detection using Deep Learning based Object Detection algorithms
• Source code: https://github.com/openalpr/openalpr, https://github.com/MicrocontrollersAndMore/OpenCV_3_License_Plate_Recognition_Cpp

• Research Paper: "A Robust Real-time Automatic License Plate Recognition based on the YOLO Detector" by J. Kim et al., 2019, https://arxiv.org/abs/1901.05079
• "Real-time vehicle license plate detection and recognition using deep learning" by M. Khan et al., 2018, https://ieeexplore.org/abstract/document/8460597/

13. Viola-Jones Algorithm

This is a classic computer vision method for detecting faces in images and video. The system uses a cascade of simple Haar-like features to efficiently detect faces in real-time.
• Codebase: https://github.com/opencv/opencv/tree/master/data/haarcascades
• Research paper: https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/viola-cvpr-01.pdf

14. Multi-task Cascaded Convolutional Networks (MTCNN)

This is a face detection system that uses cascaded convolutional networks to perform face detection, facial landmark localization, and facial attribute analysis.
• Codebase: https://github.com/kpzhang93/MTCNN_face_detection_alignment
• Research paper: https://kpzhang93.github.io/MTCNN_face_detection_alignment/paper/spl.pdf

15. Super-Resolution

This increases the resolution of an image while preserving its important details. It can be achieved using deep learning models that learn the mapping between low and high resolution images, generating a high-resolution image from a low-resolution one as output.
• Models: SRResNet, EDSR, RCAN
• Datasets: DIV2K, Set5, Set14, B100, Urban100
• Application domain: Image Processing
• Level: Intermediate
• Audience Interest level: High
• Explanation: Super-Resolution using Deep Convolutional Neural Networks
Source code:
• SRResNet: https://github.com/twtygqyy/pytorch-SRResNet

• EDSR:
https://github.com/thstkdgus35/EDSR-PyTorch
• RCAN: https://github.com/yulunzhang/RCAN
Research paper:
• SRResNet: https://arxiv.org/abs/1609.04802
• EDSR: https://arxiv.org/abs/1707.02921
• RCAN: https://arxiv.org/abs/1707.01594

16. Image Restoration

This involves removing degradation such as noise, blur, or over-exposure from an image, to enhance its quality and make it more visually appealing. This is achieved by using techniques such as denoising, deblurring, and inpainting to undo the effects of degradation and recover the original image content.
• Models: Deep Convolutional Neural Network (DCNN)
• Datasets: BSD500 dataset, DIV2K dataset
• Application domain: Image Processing
• Level: Intermediate
• Audience Interest level: High
• Explanation: Image restoration involves removing degradation and improving the visual quality of an image, such as removing noise, blurring, or removing defects.
• Source code:
https://github.com/SaoYan/Image-Restoration
• Research Paper: Image Restoration Using Convolutional Neural Networks (2017), https://arxiv.org/abs/1707.06841

17. Scene Understanding

This involves recognizing and categorizing different elements in a scene (such as objects, people, and environment) to gain a deeper understanding of the context and meaning of an image. It often involves multiple computer vision tasks, including object detection, image segmentation, and semantic segmentation.
• Models: PointNet, SceneNet RGB-D, SPADE
• Datasets: NYUv2, ScanNet, SUN RGB-D
• Application domain: 3D Scene Analysis
• Level: Intermediate
• Audience Interest level: Moderate
• Explanation: Scene Understanding using Deep Learning techniques such as PointNet, SceneNet RGB-D, and SPADE
• Source code: https://github.com/charlesq34/pointnet
Research paper:
• PointNet: https://arxiv.org/abs/1612.00593
• SceneNet RGB-D: https://arxiv.org/abs/1707.01302
• SPADE: https://arxiv.org/abs/1903.07291

18. Action Recognition

This involves identifying and classifying human actions within video data. It is commonly used in surveillance, sports analysis and human-computer interaction applications.
• Models: Two-Stream Convolutional Networks, C3D, LSTM, ResNet-50
• Datasets: UCF101, HMDB51, Kinetics
• Application domain: Video Processing
• Level: Intermediate
• Audience Interest level: High
• Explanation: Action Recognition using Deep Learning approaches to classify human actions in videos.
• Codebase:
https://github.com/axelbarroso/C3D-keras
• Research Paper:
    ◦ "Two-Stream Convolutional Networks for Action Recognition in Videos" by Simonyan and Zisserman (2014).
    ◦ "Learning Spatiotemporal Features with 3D Convolutional Networks" by Tran et al. (2015).

19. Image Style Transfer

This involves transforming an input image to have the same style as a reference image, while retaining its content. It is often achieved using Convolutional Neural Networks (CNNs) trained on large datasets of style images.
• Models: Neural Style Transfer (NST)
• Datasets: MS-COCO, ImageNet
• Application Domain: Deep Learning
• Level: Intermediate
• Audience Interest Level: High
• Explanation: Image style transfer is a task in computer vision where the style of one image is transferred to the content of another image.
• Source code:
https://github.com/cysmith/neural-style-tf
• Research Paper: A Neural Algorithm of Artistic Style, Leon A. Gatys et al. (2015)
https://arxiv.org/abs/1508.06576
****

20. Gaze Estimation

This involves predicting the direction a person is looking based on their eye and head movements. It is used to understand user attention and to provide interactive control in human-computer interaction systems.
• Models: GazeNet, DeepGaze II, MPIIGaze

• Datasets: MPIIFaceGaze, ETRA
• Application Domain: Computer Vision, Human-Computer Interaction
• Level: Beginner to Intermediate
• Audience Interest Level: Moderate
• Explanation: Gaze Estimation is the process of predicting the direction of a person's gaze based on image or video data. This can be useful for various applications, such as assistive technologies for people with disabilities, human-computer interaction, and user attention analysis.
Codebase:
• GazeNet (TensorFlow):
https://github.com/NandaMohammed/GazeNet
• DeepGaze II (MATLAB):
https://github.com/mpatacchiola/deepgaze
• MPIIGaze (MATLAB):
https://github.com/mpatacchiola/gaze-estimation
Research Papers:
• GazeNet: https://arxiv.org/abs/1710.05987
• DeepGaze II:
https://ieeexplore.org/abstract/document/7888061/
• MPIIGaze:
https://ieeexplore.org/abstract/document/7645456/

21. Image Generation

In this, we generate new images from a given input, typically a noise vector or a sample from a prior distribution. It aims to capture the underlying structure and patterns of an image dataset to generate new, diverse images that are similar in style and content to the training data.
• Models: GANs (Generative Adversarial Networks)
• Datasets: CIFAR-10, MNIST, CelebA, ImageNet
• Application domain: Generative Models, Image Processing
• Level: Intermediate
• Audience Interest level: High
• "Practical Deep Image Generation with GANs" - Code:
https://github.com/tensorflow/tensorflow/tree/master/tensorflow/contrib/eager/python/examples/generative_examples
Research Paper: https://arxiv.org/abs/1807.04720
• "DCGAN: Deep Convolutional Generative Adversarial Networks" - Code:
https://github.com/carpedm20/DCGAN-tensorflow
Research Paper: https://arxiv.org/abs/1511.06434

• "Wasserstein GAN" - Code:
https://github.com/martinarjovsky/WassersteinGAN
Research Paper: https://arxiv.org/abs/1701.07875
• "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks" - Code:
https://github.com/eriklindernoren/PyTorch-GAN
Research Paper: https://arxiv.org/abs/1511.06434
• "Progressive Growing of GANs for Improved Quality, Stability, and Variation" - Code:
https://github.com/tkarras/progressive_growing_of_gans
Research Paper: https://arxiv.org/abs/1710.10196
Note: These are just some popular examples of Image Generation projects and many more models exist in this domain, some of which are variations of the above mentioned models.

22. Image Captioning

This is a task in computer vision where a textual description is generated for an input image, aimed at explaining its content to a human reader. It uses deep learning models to learn the mapping between image and textual representations.
• Models: Encoder-Decoder Models, Attention-based Models
• Datasets: Microsoft COCO, Flickr30k
• Application Domain: Natural Language Processing
• Level: Intermediate
• Audience Interest Level: High
• Explanation: Image Captioning involves generating textual description of an image. It is a task in computer vision and natural language processing, and is typically performed using Encoder-Decoder models or Attention-based models.
Codebase:
• PyTorch Tutorial on Image Captioning:
https://github.com/yunjey/pytorch-tutorial/tree/master/tutorials/03-advanced/image_captioning
• TensorFlow Tutorial on Image Captioning:
https://github.com/tensorflow/models/tree/master/research/im2txt
Research Paper:
• "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention" (2015) by Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel: https://arxiv.org/abs/1502.03044
• "Are Neural Image Captions Better than Humans at Describing Images?" (2017) by Ting Yao, Yaoxiong Li, Fan Ma, Jianping Shen, Zhiheng Huang, Wei Liu, Ting Liu:
https://arxiv.org/abs/1705.04875

23. Disease detection CV projects

Here are some disease detection projects in Computer Vision and their codebase/research paper links:
• Plant Disease Detection:
Codebase: https://github.com/hardikvasa/image-classification-keras
Research Paper: https://ieeexplore.org/abstract/document/8926551
• Skin Lesion Analysis for Melanoma Detection:
Codebase: https://github.com/parhambarazesh/Skin-Lesion-Analysis
Research Paper: https://arxiv.org/abs/1907.03380
• Diabetic Retinopathy Detection:
Codebase: https://github.com/EyePainter/Diabetic-Retinopathy-Detection
Research Paper: https://www.sciencedirect.com/science/article/pii/S2213177917304780
• Pneumonia Detection using Chest X-rays:
Codebase: https://github.com/akshaybahadur21/Pneumonia-Detection
Research Paper: https://ieeexplore.ieee.org/document/8869364
• Lung Nodule Detection and Classification:
Codebase: https://github.com/tanaymukherjee/Lung-Nodule-Detection-Classification
Research Paper: https://ieeexplore.org/abstract/document/8661197

==

Forecasting projects

Sl. No.	TITLE	Dataset Link	Code
1	Cirrhosis Patient Survival Prediction	cirrhosis.csv	Cirrhosis Patient Survival Prediction | Kaggle
2	Life Insurance Prediction (Predicting if a person would buy life insurance based on his age using logistic regression)	life_insurance_data.csv	Life Insurnace Prediction (Logistic Regression) | Kaggle
3	Employee Attrition Prediction	Employee-Attrition.csv	Employee_Churn_Prediction | Kaggle
4	Air Pollution Forecasting	AirPollution.csv	GitHub - jyoti0225/Air-Pollution-Forecasting: Time Series Analysis of Air Pollutants(PM2.5) using LSTM model
5	Bitcoin Price prediction	bitcoin.csv	GitHub - Pradnya1208/Bitcoin-Price-Prediction-using-ARIMA: Bitcoin price prediction using ARIMA Model.
6	Predict Car Prices using PyTorch	CarPrice.csv	Predict Car Prices with Machine Learning | Thecleverprogrammer
7	House Price prediction using regression	house_price.csv	GitHub - sharmaji27/House-Price-Prediction-USA-Housing-Data
8	Online Shoppers Behavior Prediction	online_shoppers_intention.csv	Online Shoppers Behavior Prediction | Kaggle
9	Weather-prediction	weather.csv	Weather-prediction (Naive Bayes-Gaussian) | Kaggle
10	Hotel Booking Prediction	hotel_bookings.csv	Hotel Booking Data Analysis - Case Study | Kaggle

11	Energy Demand Forecasting	energy_demand.csv	Time Series Forecasting with XGBoost | Kaggle
12	Traffic Volume Prediction	traffic.csv	GitHub - xiaochus/TrafficFlowPrediction: Traffic Flow Prediction with Neural Networks(SAEs、LSTM、GRU).
13	Earthquake Prediction Model	earthquake.csv	Earthquake Prediction Model with Machine Learning | Thecleverprogrammer
14	Rainfall Prediction using Logistic Regression	rainfall.csv	Rainfall Prediction with Machine Learning | Thecleverprogrammer
15	Rainfall Prediction using Decision Tree	rainfall.csv	Rainfall Prediction with Machine Learning | Thecleverprogrammer
16	Rainfall Prediction using Neural Network	rainfall.csv	Rainfall Prediction with Machine Learning | Thecleverprogrammer
17	Rainfall Prediction using Random Forest	rainfall.csv	Rainfall Prediction with Machine Learning | Thecleverprogrammer
18	Rainfall Prediction using LightGBM	rainfall.csv	Rainfall Prediction with Machine Learning | Thecleverprogrammer
19	Rainfall Prediction using Catboost	rainfall.csv	Rainfall Prediction with Machine Learning | Thecleverprogrammer
20	Rainfall Prediction using XGBoost	rainfall.csv	Rainfall Prediction with Machine Learning | Thecleverprogrammer

==

Speech and NLP projects:

1. **Tokenization:**
   Dataset:**Tokenization dataset**
   Codebook : **Basics of nlp**

2. **Text summarization:**
   Dataset: **Text summarization dataset**
   Codebook : **Text summarization example codebase**

3. **Pos- tagging:**
   Dataset: **Pos tagging dataset**
   Codebook : **Pos tagging using RNN**

4. **Email spam filter:**
   Dataset: **Email spam dataset**
   Codebook : **Kaggle notebook for email spam classification**

5. **Hate Speech classification :**
   Dataset: **Hate speech dataset**
   Codebook : **Twitter hate speech classifier**

6. **Twitter sentiment analysis:**
   Dataset: **Twitter sentiment analysis dataset**
   Codebook : **Twitter sentiment analysis**

7. **MFCC extraction:**
   Dataset: **MFCC extraction dataset**
   Codebook : **MFCC feature extraction from audio**

8. **MFCC + KNN for Frog sound detection and classification:**
   Dataset: **MFCC classification**
   Codebook:**Feature extraction and KNN**

9. **MFCC + Naive bayes for Frog sound detection and classification:**
   Dataset: **MFCC classification**
   Codebook:

10. **MFCC + CNN for Frog sound detection and classification:**
    Dataset: **MFCC classification**
    Codebook:**Audio classification using CNN**

11. **Speech emotion recognition using KNN:**
    Dataset: **Speech Emotion Recognition dataset**
    Codebook: **Speech emotion recognition using KNN**

12. **Speech emotion recognition using MLP:**
    Dataset: **Speech Emotion Recognition dataset**
    Codebook:**Speech emotion recognition using diff algo**

13. **Speech emotion recognition using CNN:**
    Dataset: **Speech Emotion Recognition dataset**
    Codebook:**Speech emotion recognition using diff algo**

14. **Music genre classification using KNN:**
    Dataset: **Music genre classification**
    Codebook:**Music genre classification using KNN**

15. **Music genre classification using CNN:**

Dataset: **Music genre classification**

Codebook: **Music genre classification using CNN**