COMPUTER VISION

ROADMAP

ULTIMATE COMPUTER VISION ROADMAP – THE BEST EVER!

Brought to you by Path2Proficiency – Your ultimate destination for mastering next-level skills!

Goal: Master Computer Vision (CV) to develop object detection models, face recognition systems, real-time video analysis, and more using OpenCV, TensorFlow, PyTorch, and advanced AI models.

Why This Roadmap?

- Covers **Basics to Advanced Techniques** with real-world applications.
- Learn OpenCV, TensorFlow, and PyTorch with hands-on projects.
- Build and deploy custom computer vision models.
- Covers MLOps, Model Deployment, and Computer Vision in Production.

Phase 1: Introduction to Computer Vision

Goal: Build a strong foundation in computer vision concepts.

1. Basics of Image Processing

Topics to Cover:

- What is Computer Vision? Applications and Real-World Use Cases
- Basics of Image Representation (RGB, Grayscale, Pixels)
- Image Manipulation Techniques (Cropping, Resizing, Rotation)
- Histogram Equalization and Image Thresholding

References:

- Introduction to Computer Vision – OpenCV
- Python for Computer Vision – FreeCodeCamp

Hands-On Task:

- Load and display images using OpenCV.
- Perform basic image transformations and save results.

2. Image Filtering and Transformation Techniques

Topics to Cover:

- Convolution and Kernels
- Edge Detection (Sobel, Canny)
- Image Smoothing and Blurring
- Morphological Operations (Erosion, Dilation)

References:

- Image Filtering Basics – OpenCV Docs
- Canny Edge Detection – Sentdex

Hands-On Task:

- Implement Canny Edge Detection and apply filters.
- Perform morphological operations on sample images.

Phase 2: Feature Extraction and Object Detection

Goal: Detect, classify, and extract features from images.

3. Feature Detection and Description

Topics to Cover:

- Corner Detection (Harris, Shi-Tomasi)
- SIFT, SURF, ORB Feature Detectors
- Keypoint Matching and Feature Descriptors

References:

- Feature Detection in OpenCV – GeeksForGeeks
- SIFT and SURF Tutorial – Analytics Vidhya

Hands-On Task:

- Perform feature detection using SIFT and ORB.
- Match keypoints between two images using Brute-Force Matcher.

4. Object Detection using Classical Algorithms

Topics to Cover:

- Template Matching
- Haar Cascades for Face Detection
- HOG (Histogram of Oriented Gradients)

References:

- Object Detection with OpenCV – PyImageSearch
- Face Detection using Haar Cascades

Hands-On Task:

- Detect faces using Haar Cascades.
- Implement HOG-based object detection.

Phase 3: Deep Learning for Computer Vision

Goal: Build and train deep learning models for computer vision tasks.

5. Introduction to Neural Networks for Vision

Topics to Cover:

- Basics of CNN (Convolutional Neural Networks)
- CNN Architecture – Convolution, Pooling, Fully Connected Layers
- Understanding ReLU, MaxPooling, and Softmax

References:

- Deep Learning for CV – Stanford CS231n
- CNN Basics – Sentdex

Hands-On Task:

- Build a CNN for image classification using TensorFlow/Keras.
- Train the CNN on the MNIST dataset.

6. Image Classification and Transfer Learning

Topics to Cover:

- Image Classification with CNNs
- Transfer Learning with Pretrained Models (VGG, ResNet, Inception)
- Fine-Tuning Pretrained Models

References:

- Transfer Learning with Keras – TensorFlow Docs
- Fine-Tuning CNN Models – PyTorch Docs

Hands-On Task:

- Implement Transfer Learning using VGG/ResNet.
- Fine-tune a CNN model for a custom dataset.

7. Object Detection with Deep Learning

Topics to Cover:

- Introduction to Object Detection Models

- YOLO (You Only Look Once) – Basics and Applications
- SSD (Single Shot Detector) and Faster R-CNN

References:

- YOLO Basics – PyImageSearch
- SSD and Faster R-CNN Explained – Analytics Vidhya

Hands-On Task:

- Implement YOLOv5 for real-time object detection.
- Train a Faster R-CNN on a custom dataset.

Phase 4: Advanced Computer Vision Techniques

Goal: Master advanced concepts like segmentation, GANs, and facial recognition.

8. Image Segmentation and Instance Segmentation

Topics to Cover:

- Semantic Segmentation using U-Net
- Instance Segmentation using Mask R-CNN
- Object Segmentation Techniques

References:

- U-Net for Image Segmentation – Medium
- Mask R-CNN for Object Segmentation

Hands-On Task:

- Implement U-Net for semantic segmentation.
- Use Mask R-CNN for instance segmentation.

9. Generative Models and GANs for Vision

Topics to Cover:

- Introduction to GANs (Generative Adversarial Networks)
- DCGAN, CycleGAN, and StyleGAN
- Image Synthesis and Style Transfer

References:

- GANs for Image Generation – Medium
- CycleGAN and Style Transfer – PyTorch Docs

Hands-On Task:

- Generate new images using DCGAN.

- Implement CycleGAN for image-to-image translation.

Phase 5: Face Recognition, Tracking, and Pose Estimation

Goal: Build real-world applications using facial and body recognition.

10. Face Recognition and Emotion Detection

Topics to Cover:

- Face Detection using MTCNN
- Face Recognition with FaceNet/Dlib
- Emotion and Expression Detection

References:

- Face Recognition Basics – PyImageSearch
- Emotion Detection using CNN – Analytics Vidhya

Hands-On Task:

- Build a face recognition system using Dlib.
- Create an emotion recognition system with CNN.

11. Pose Estimation and Human Activity Recognition

Topics to Cover:

- OpenPose and MediaPipe Basics
- 2D and 3D Pose Estimation Models
- Action Recognition using LSTM/CNN

References:

- Pose Estimation with OpenPose – Medium
- MediaPipe for Pose Estimation

Hands-On Task:

- Implement pose estimation using MediaPipe.
- Train a human activity recognition model.

Phase 6: MLOps, Model Deployment & Scalability

Goal: Deploy and manage CV models in production environments.

12. Model Deployment & API Integration

Topics to Cover:

- Deploying Models using Flask/Django

- Building REST APIs for Computer Vision Models
- Model Versioning and Monitoring

References:

- Deploying CV Models with Flask – Codebasics
- REST API with FastAPI – FastAPI Docs

Hands-On Task:

- Deploy a Flask API for object detection.
- Build a FastAPI-based face recognition API.

13. MLOps & Model Management for CV

Topics to Cover:

- CI/CD for CV Models
- Dockerizing Computer Vision Models
- Kubernetes and Model Orchestration

References:

- MLOps for CV Models – Coursera
- Docker Basics for AI – Docker Docs

Hands-On Task:

- Build a CI/CD pipeline for CV models.
- Deploy a scalable CV API using Kubernetes.

Phase 7: Capstone Projects & Portfolio Building

Goal: Build and showcase advanced CV projects to highlight your expertise.

14. Real-World Capstone Project Ideas

Project Ideas:

- **Real-Time Face Mask Detection System**
- **AI-Powered Object Detection for Autonomous Vehicles**
- **Human Pose Estimation for Fitness Tracking**
- **Handwritten Digit Recognition System**

Video Surveillance System with Real-Time Alerts

Estimated Timeline to Master Computer Vision:

Beginner to Intermediate: 3-4 months
Intermediate to Advanced: 5-6 months
Deployment & MLOps: 2-3 months

By Following This Roadmap, You Will:

Build and Deploy State-of-the-Art CV Models.
Master CNNs, YOLO, GANs, and Face Recognition.
Become a Computer Vision Expert Ready for Production Challenges.

Powered by Path2Proficiency – Where Learning Meets Mastery!