



**COMPUTER VISION: ARCHITECTURAL PARADIGMS,
OPERATIONAL PIPELINES, AND INDUSTRIAL
APPLICATIONS
(GROUP XII)**

COURSE TITLE: SPECIAL TOPICS IN COMPUTER SCIENCE

COURSE CODE: CMP 416

LECTURER: DR. DEBORAH ALEBURU

Table of Content

Group Members	4
1. INTRODUCTION & DEFINITION	5
1.1. What is Computer Vision?	5
1.2. Core Objectives	5
1.3. Relationship to Related Fields	5
2. HISTORICAL EVOLUTION & TIMELINE	7
2.1 Early Foundations (1950s-1960s)	7
2.2 Development Era (1970s-1980s)	7
2.3 Advancement Era (1990s-2000s)	7
2.4 Deep Learning Revolution (2010-Present)	8
3. FUNDAMENTALS & CORE CONCEPTS	9
3.1 How Computer Vision Works: The Complete Pipeline	9
3.2 Key Components of Computer Vision Systems	10
4. Core Architectures: The Engines of Vision	12
4.1 Convolutional Neural Networks (CNNs)	12
4.1.1 Architectural Mechanics	12
4.1.2 Landmark Architectures	12
4.2 Vision Transformers (ViTs)	13
4.2.1 The ViT Pipeline	13
4.2.2 Comparative Analysis: CNN vs. ViT	13
4.3 Advanced Object Detection: The YOLO Series	14
4.3.1 YOLOv11 (2024/2025)	14
4.4 Segmentation and Foundation Models: SAM 2	15
5. REAL-WORLD APPLICATIONS BY INDUSTRY	16
5.1 Healthcare & Medical Imaging	16
5.2 Autonomous Vehicles	16
5.3 Retail & E-commerce	17
5.4 Manufacturing & Quality Control	18
5.5 Agriculture	18
5.6 Security & Surveillance	19
5.7 Sports Analytics	20
5.8 Transportation & Logistics	20
5.9 Document Processing	21
5.10 Smart Cities	22
6. CHALLENGES & LIMITATIONS	23
6.1 Data-Related Challenges	23

6.2 Technical Challenges	23
6.3 Computational Challenges	24
6.4 Robustness & Generalization	25
6.5 Privacy & Ethical Concerns	25
6.6 Interpretability & Explainability	26
7. FUTURE TRENDS & INNOVATIONS	27
7.1 Vision Transformers	27
7.2 Multimodal Learning	27
7.3 Self-Supervised Learning	27
7.4 Few-Shot & Zero-Shot Learning	27
7.5 3D Scene Understanding	27
7.6 Efficient & Edge AI	28
7.7 Continual Learning	28
7.8 Embodied AI & Robotics	28
8. GETTING STARTED WITH COMPUTER VISION	29
9. Conclusion	30
10. REFERENCES	31

Group Members

1. Ajayi Oluwapelumi - AUL/CMP/22/013
2. Raphael Fulfilled - AUL/CMP/22/080
3. Oluwadare Penuel - AUL/CMP/22/096
4. Orilade Richard - AUL/CMP/23/074
5. Iyanda Jeremiah - AUL/CMP/23/093

1. INTRODUCTION & DEFINITION

1.1. What is Computer Vision?

Computer Vision is the science and technology that enables machines to understand, analyze, and interpret visual information in the same way humans do. Unlike human vision, which evolved over millions of years, computer vision systems use algorithms, sensors, and computational methods to extract meaningful information from images and videos.

1.2. Core Objectives

- Image Understanding: Teaching computers to perceive and interpret visual data
- Feature Recognition: Identifying and extracting important patterns from images
- Object Detection & Recognition: Locating and classifying objects in images
- Scene Interpretation: Understanding spatial relationships and context
- Decision Making: Making informed decisions based on visual information

1.3. Relationship to Related Fields

Computer Vision differs from but is closely related to several other fields:

Field	Focus	Scope
Computer Vision	Understanding and interpreting visual data broadly	Broad applications across industries
Image Processing	Enhancing and transforming images (filtering, noise removal)	Lower-level image enhancement
Machine Vision	Quality control and measurement in industrial settings	Narrow, application-specific tasks

Image Analysis	Extracting quantitative information from images	Measurement and quantification
---------------------------------	--	-----------------------------------

2. HISTORICAL EVOLUTION & TIMELINE

2.1 Early Foundations (1950s-1960s)

Key Milestone: The Birth of Neural Networks

1950s: Perception and shape recognition experiments begin

1958: Frank Rosenblatt invents the Perceptron, an early form of neural network designed for pattern recognition. This laid the foundational stone for modern deep learning approaches.

1963: Computers become capable of interpreting three-dimensional objects from 2D images. Artificial Intelligence becomes an established academic discipline.

1966: MIT establishes the Artificial Intelligence Laboratory. Researchers attempt the ambitious "Summer Vision Project," trying to build a complete vision system—showing the complexity of the task.

2.2 Development Era (1970s-1980s)

Key Achievements: Recognition & Pattern Detection

1974: Optical Character Recognition (OCR) is introduced, enabling computers to interpret text in any typeface. This was a major breakthrough for document processing.

1980: Dr. Kunihiro Fukushima proposes the Neocognitron, a hierarchical multilayered neural network capable of robust visual pattern recognition. It could detect:

- Corners and curves
- Edges
- Basic shapes
- Complex patterns

2.3 Advancement Era (1990s-2000s)

Key Achievements: Real-Time Processing & Face Recognition

1990s-2000s: Development of face recognition algorithms and dramatic improvements in real-time image processing speed

2000-2001: Studies on object recognition increase significantly, leading to the first real-time face recognition applications

2001: Viola-Jones algorithm for rapid face detection is developed, enabling cascade classifiers

2.4 Deep Learning Revolution (2010-Present)

Key Achievements: CNNs & Modern Systems

2010: ImageNet dataset becomes available containing millions of tagged images across various object classes. This provides the foundation for CNN development.

2012: AlexNet wins the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), demonstrating the power of deep convolutional neural networks. This marks the beginning of the deep learning revolution in computer vision.

2014: COCO (Common Objects in Context) dataset is developed for object detection and instance segmentation

VGGNet and GoogleNet are introduced

Faster R-CNN is developed for object detection

2015: ResNet wins ImageNet with 152 layers, proving that very deep networks can be effectively trained

2016: YOLO (You Only Look Once) v2 advances real-time object detection

2017: Vision Transformers and attention mechanisms begin to gain prominence

2020-Present: State-of-the-art models achieve superhuman accuracy on many benchmarks

3. FUNDAMENTALS & CORE CONCEPTS

3.1 How Computer Vision Works: The Complete Pipeline

Computer vision systems process visual data through a systematic pipeline:

Step 1: Image Acquisition

- Cameras or sensors capture visual data
- Data can be single images or video sequences
- Sources: digital cameras, mobile devices, medical imaging equipment, satellites
- Formats: 2D images, 3D point clouds, video sequences

Step 2: Image Preprocessing

Preparing raw data for analysis:

- Noise Reduction: Removing unwanted artifacts (Gaussian blur, median filtering)
- Image Enhancement: Improving contrast and brightness (histogram equalization)
- Normalization: Scaling pixel values to standard ranges
- Resizing: Adjusting image dimensions for consistent processing
- Color Space Conversion: Converting between RGB, HSV, Grayscale, etc.
- Distortion Correction: Correcting lens distortion and perspective issues

Step 3: Feature Detection & Extraction

Identifying key patterns and features:

- Edge Detection (Canny, Sobel, Laplacian)
- Corner Detection (Harris, Shi-Tomasi)
- Blob Detection: Finding regions with distinct properties
- Feature Descriptors: SIFT, SURF, ORB (creating representations of detected features)

- Texture Analysis: Identifying surface patterns and properties

Step 4: Feature Description & Matching

- Creating mathematical descriptions of detected features
- Matching features across multiple images
- Building feature correspondence maps
- Enabling feature-based tracking and alignment

Step 5: Pattern Recognition & Classification

Using learned patterns to identify objects:

- Traditional ML: Support Vector Machines (SVMs), k-Nearest Neighbors
- Deep Learning: Convolutional Neural Networks (CNNs)
- Template Matching: Comparing image regions to predefined templates

Step 6: Post-Processing & Decision Making

- Refining detected objects and classifications
- Applying morphological operations (erosion, dilation)
- Filtering false positives
- Making final decisions or taking actions based on results

3.2 Key Components of Computer Vision Systems

Image Representation

- Pixels: Fundamental units with intensity/color values
- Channels: RGB (Red, Green, Blue), Grayscale, HSV, etc.
- Resolution: Image dimensions (width \times height in pixels)
- Bit Depth: Color information per pixel (8-bit, 16-bit, 32-bit)

Mathematical Foundations

- Linear Algebra: Matrix operations for image transformations

- Probability & Statistics: Uncertainty quantification, Bayesian inference
- Calculus: Optimization for training neural networks
- Geometry: 3D reconstruction, camera calibration, perspective transforms

Hardware & Acceleration

- GPUs: NVIDIA CUDA for parallel processing
- TPUs: Tensor Processing Units for specialized AI workloads
- Edge Devices: Mobile processors (ARM, Qualcomm Snapdragon)
- FPGA: Field-programmable gate arrays for custom hardware acceleration

4. Core Architectures: The Engines of Vision

The choice of architecture dictates the performance, speed, and capability of a computer vision system. In 2025, the landscape is dominated by three main families: Convolutional Neural Networks (CNNs), Vision Transformers (ViTs), and hybrid models.

4.1 Convolutional Neural Networks (CNNs)

CNNs are biologically inspired architectures designed to process grid-like data. They rely on the concept of *local receptive fields* and *weight sharing*.

4.1.1 Architectural Mechanics

The standard CNN pipeline consists of three primary layer types:

- a. **Convolutional Layers:** These layers apply a set of learnable filters (kernels) to the input. As a filter slides (convolves) across the image, it performs element-wise multiplication and summation, generating a feature map. This allows the network to detect local patterns like edges regardless of their position in the image (translation invariance).
- b. **Pooling Layers:** Pooling (typically Max Pooling) reduces the spatial dimensions of the feature maps. This downsampling reduces computational complexity and makes the representation invariant to small translations and distortions.
- c. **Fully Connected (Dense) Layers:** At the end of the network, the high-level feature maps are flattened into a vector and passed through dense layers to perform the final classification or regression.

4.1.2 Landmark Architectures

- **ResNet (Residual Networks):** Introduced "skip connections" that allow gradients to flow through the network without vanishing. This innovation enabled the training of extremely deep networks (100+ layers), which is critical for capturing complex semantic features.
- **EfficientNet:** Utilized "compound scaling" to uniformly scale network width, depth, and resolution, achieving better performance with fewer parameters than previous models.

4.2 Vision Transformers (ViTs)

Vision Transformers represent a paradigm shift from local processing (CNNs) to global processing. Originally designed for Natural Language Processing (NLP), Transformers use "self-attention" to weigh the importance of different parts of an image relative to each other.

4.2.1 The ViT Pipeline

1. **Patching:** The input image is divided into fixed-size square patches (e.g., 16×16 pixels).
2. **Linear Embedding:** Each patch is flattened and projected into a linear vector (embedding), analogous to a word token in NLP.
3. **Positional Encoding:** Since Transformers process patches in parallel and lack inherent knowledge of spatial structure, learnable positional vectors are added to the patch embeddings to retain location information.
4. **Self-Attention:** The model computes the relationship between every patch and every other patch. This allows the network to capture long-range dependencies across the entire image understanding the context of a pixel in the top-left corner relative to one in the bottom-right.

4.2.2 Comparative Analysis: CNN vs. ViT

Feature	Convolutional Neural Networks (CNNs)	Vision Transformers (ViTs)
Inductive Bias	Strong (Translation invariance, locality)	Weak (Must learn relationships from data)
Data Requirement	High efficiency on smaller datasets	Requires massive datasets to outperform CNNs

Receptive Field	Local (grows with depth)	Global (immediate access to whole image)
Robustness	Sensitive to texture/noise	Robust to occlusions and permutations
Training Cost	Lower	Higher (Quadratic complexity with tokens)

4.3 Advanced Object Detection: The YOLO Series

For real-time applications, the You Only Look Once (YOLO) architecture is the industry standard. Unlike two-stage detectors (like Faster R-CNN) that first propose regions and then classify them, YOLO frames detection as a single regression problem, predicting bounding boxes and probabilities directly from full images.

4.3.1 YOLOv11 (2024/2025)

The latest iteration, YOLOv11, introduces significant architectural refinements over YOLOv8:

C3k2 Block: An optimized feature extraction module that improves efficiency.

C2PSA (Cross Stage Partial with Spatial Attention): A mechanism that enhances the model's ability to focus on specific areas of interest, improving detection of small objects and handling complex backgrounds.

Performance: YOLO11 achieves higher mean Average Precision (mAP) on benchmarks like COCO while utilizing 22% fewer parameters than YOLOv8, making it ideal for edge deployment on devices like Raspberry Pi or Jetson Nano.

4.4 Segmentation and Foundation Models: SAM 2

The Segment Anything Model 2 (SAM 2) by Meta represents the cutting edge of segmentation.

Video Segmentation: SAM 2 extends image segmentation to video by introducing a memory mechanism (a memory bank) that stores information about objects across frames. This allows the model to track objects consistently even when they are temporarily occluded.

Promptability: Users can interact with the model by clicking, drawing boxes, or providing text prompts, making it a highly versatile tool for interactive annotation and editing.

5. REAL-WORLD APPLICATIONS BY INDUSTRY

5.1 Healthcare & Medical Imaging

Current Applications:

- Medical Image Analysis: MRI, CT scan, X-ray interpretation
- Tumor Detection: Cancer screening and diagnosis acceleration
- Disease Classification: Identifying pathological conditions
- Surgical Guidance: Real-time assistance during procedures

Impact:

- Faster, more accurate diagnoses
- Early disease detection
- Reduced human error
- Improved patient outcomes

Technologies Used:

- U-Net, DeepLab (segmentation)
- ResNet, EfficientNet (classification)
- 3D CNNs (volumetric data)

5.2 Autonomous Vehicles

Critical Computer Vision Systems:

Object Detection:

- Detect pedestrians, vehicles, cyclists
- Identify road signs and traffic lights
- Locate lane markings

Semantic Segmentation:

- Identify drivable surfaces
- Distinguish obstacles

- Understand road context

3D Detection & Tracking:

- Localize objects in 3D space
- Predict object trajectories
- Avoid collisions

Performance Requirements:

- Real-time processing (30+ fps)
- High reliability and safety
- Robustness to adverse weather

Architectures: YOLO, Faster R-CNN, PointNet (for 3D)

5.3 Retail & E-commerce

Computer Vision Applications:

Product Recognition:

- Identify products from images
- Recommend similar items
- Automated catalog management

Customer Analytics:

- Heatmaps tracking customer movement
- Dwell time analysis
- Queue monitoring

Cashier-Less Stores:

- Real-time inventory tracking
- Fraud detection
- Automated checkout via facial recognition

Inventory Management:

- Stock level monitoring
- Product placement optimization
- Loss prevention

Technologies: Object detection, segmentation, pose estimation

5.4 Manufacturing & Quality Control

Automated Inspection Systems:

Defect Detection:

- Surface defects (scratches, dents, discoloration)
- Dimensional accuracy verification
- Component placement verification

Process Monitoring:

- Assembly line quality assurance
- Production efficiency optimization
- Predictive maintenance

3D Manufacturing:

- Hyper-detailed component assembly
- Laser/rotary die cutting precision
- Automated material handling

Benefits:

- 99.9%+ detection accuracy
- Elimination of human inspection variability
- Real-time production adjustments

Architectures: CNNs, custom U-Net variants

5.5 Agriculture

Precision Farming Applications:

Crop Health Monitoring:

- Detect plant diseases
- Identify nutrient deficiencies
- Monitor crop growth
- Assess irrigation needs

Pest & Weed Management:

- Identify invasive species
- Guide targeted pesticide application
- Monitor pest populations

Yield Prediction:

- Estimate harvest quantity
- Optimize harvesting schedules
- Improve resource allocation

Implementation: Aerial imagery, drone monitoring, field sensors

5.6 Security & Surveillance

Smart Surveillance Systems:

Threat Detection:

- Person detection and tracking
- Weapon/contraband detection
- Unusual behavior identification

Biometric Authentication:

- Facial recognition
- Iris scanning
- Gait analysis

Access Control:

- Real-time identity verification
- Intrusion detection
- Multi-person tracking

Privacy & Ethics Considerations:

- Consent and transparency
- Bias in facial recognition
- Data protection regulations (GDPR, etc.)

5.7 Sports Analytics

Computer Vision for Sports:

Player Tracking:

- Multi-player tracking across frames
- Movement pattern analysis
- Performance metrics calculation

Ball Tracking:

- Predict ball trajectory
- Detect key moments
- Instant replay analysis

Action Recognition:

- Identify specific plays
- Coach feedback generation
- Game statistics automation

Applications: Broadcasting enhancement, coaching analytics, fantasy sports

5.8 Transportation & Logistics

Fleet Management:

Vehicle Monitoring:

- Driver behavior analysis
- Accident detection
- Route optimization

Cargo Tracking:

- Shipment scanning and tracking
- Damage detection
- Automated sorting

License Plate Recognition:

- Toll collection
- Traffic enforcement
- Parking management

5.9 Document Processing

OCR & Document Understanding:

Document Scanning:

- Edge detection and document alignment
- Automatic perspective correction
- Quality enhancement

Form Processing:

- Automated data extraction
- Field recognition and classification
- Handwriting recognition

Applications:

- Adobe Scan, CamScanner
- Invoice processing
- Legal document analysis

5.10 Smart Cities

Urban Infrastructure:

Traffic Management:

- Vehicle counting and classification
- Congestion detection
- Signal optimization

Public Safety:

- Pedestrian monitoring
- Crowd analysis
- Emergency response optimization

Infrastructure Inspection:

- Bridge and road condition monitoring
- Utility network assessment
- Maintenance scheduling

6. CHALLENGES & LIMITATIONS

6.1 Data-Related Challenges

Insufficient Training Data:

- Deep learning typically requires millions of labeled images
- Expensive and time-consuming to label manually
- Mitigation: Transfer learning, data augmentation, synthetic data

Data Imbalance:

- Unequal distribution of classes
- System biased toward majority classes

Example: 99% normal images, 1% cancer

Solutions: Class weighting, oversampling, SMOTE

Annotation Quality:

- Crowdsourcing leads to inconsistent labels
- Expensive expert annotation for specialized domains
- Ambiguity in label definitions

6.2 Technical Challenges

Small Object Detection:

- Objects occupying few pixels in image
- Limited feature information
- Challenges with most detectors (YOLO, SSD)

Solutions: Multi-scale feature pyramids, attention mechanisms

Occlusion & Clutter:

- Objects partially hidden behind other objects
- Crowded scenes with overlapping objects

- Requires reasoning about occluded parts

Lighting Variations:

- Shadows, reflections, extreme brightness
- Different lighting conditions in training vs. deployment
- Affects edge and feature detection

Scale Variations:

- Objects at vastly different scales
- Single network must handle all scales
- Solution: Multi-scale architectures, feature pyramids

Geometric Transformations:

- Rotation, perspective distortion, skew
- Limited rotation invariance in CNNs
- Solutions: Data augmentation, rotation-invariant descriptors

6.3 Computational Challenges

Real-Time Performance Requirements:

- Applications like autonomous driving need <33ms inference
- Edge devices have limited computational power
- Trade-off between accuracy and speed

Energy Consumption:

- Large models require high power
- Problematic for mobile and embedded systems
- Solution: Model compression, quantization, efficient architectures

GPU Memory Limitations:

- Large batch sizes needed for training
- Limited GPU memory (12GB typical)

- Solution: Gradient checkpointing, mixed precision

6.4 Robustness & Generalization

Adversarial Examples:

- Imperceptible perturbations cause misclassification
- Highlights model brittleness
- Security concern for safety-critical applications

Domain Shift:

- Models trained on one domain perform poorly on another
- Example: Model trained on clear weather, tested in rain/snow
- Requires domain adaptation techniques

Out-of-Distribution Generalization:

- Model sees only specific object categories in training
- Struggles with novel categories at test time
- Human vision generalizes much better

6.5 Privacy & Ethical Concerns

Facial Recognition Bias:

- Higher error rates on darker skin tones
- Gender and age biases observed
- Perpetuates discrimination
- Requires careful dataset curation and fairness metrics

Surveillance & Privacy:

- Mass surveillance capabilities raise privacy concerns
- Potential for tracking and monitoring without consent
- Legal and ethical regulations needed

Data Privacy:

- Training data may contain sensitive personal information
- Privacy-preserving techniques (differential privacy, federated learning)
- GDPR and similar regulations

6.6 Interpretability & Explainability

Black Box Nature of Deep Learning:

- Difficult to understand why network makes specific predictions
- Critical for healthcare, legal, and safety-critical applications

Visualization Techniques:

- Grad-CAM: Highlight important image regions
- Feature visualization: Show what filters learn
- Saliency maps: Show pixel importance

7. FUTURE TRENDS & INNOVATIONS

7.1 Vision Transformers

- Attention-based architecture replacing CNNs
- ViT (Vision Transformer) uses transformer blocks
- Advantages: Better long-range dependencies, scalable
- Challenge: Require more training data than CNNs

7.2 Multimodal Learning

- Combining vision with other modalities
- Vision + Language (image captioning, VQA)
- Vision + Audio (video understanding)
- Foundation models (CLIP, BLIP)

7.3 Self-Supervised Learning

- Learning from unlabeled data
- Pre-training on massive unlabeled datasets
- Fine-tuning on small labeled datasets
- Reduces dependency on expensive annotations

7.4 Few-Shot & Zero-Shot Learning

- Learning from very few examples
- Meta-learning approaches
- Transfer learning to novel categories
- Closer to human learning efficiency

7.5 3D Scene Understanding

- Complete scene representation (geometry, appearance, semantics)
- Neural Radiance Fields (NeRF)
- Point cloud processing (PointNet)

- Volumetric representations

7.6 Efficient & Edge AI

- Model compression (pruning, quantization, distillation)
- Lightweight architectures (MobileNet, EfficientNet)
- On-device inference for privacy and speed
- TensorFlow Lite, ONNX Runtime

7.7 Continual Learning

- Learning from streaming data
- Adapting to distribution shifts
- Avoiding catastrophic forgetting
- Lifelong learning systems

7.8 Embodied AI & Robotics

- Vision for robotic perception and control
- Real-world generalization
- Sim-to-real transfer
- Active learning and visual exploration

8. GETTING STARTED WITH COMPUTER VISION

8.1. Essential Tools & Libraries

Python Libraries:

- OpenCV: Traditional CV algorithms, image processing
- TensorFlow/Keras: Deep learning framework by Google
- PyTorch: Deep learning framework by Facebook, preferred for research
- scikit-image: Advanced image processing algorithms
- scikit-learn: Machine learning algorithms for traditional approaches

Frameworks for Model Training:

- Detectron2: Facebook's detection framework
- MMDetection: Comprehensive object detection toolbox
- YOLOv5: Ultralytics implementation of YOLO
- TorchVision: PyTorch computer vision utilities

Public Datasets:

- ImageNet: 14M images, 1,000 object classes
- COCO: 330K images, 91 object categories, instance segmentation
- PASCAL VOC: 11,530 images, 20 object classes
- Cityscapes: Urban driving scenes with semantic segmentation
- Medical datasets: Radiology databases (CheXpert, MICCAI challenges)

Learning Path

- Foundation: Image processing basics, filters, edge detection
- Classical ML: Feature detection (SIFT, ORB), classification algorithms
- Deep Learning: CNNs, transfer learning, pre-trained models
- Applications: Choose domain (medical, autonomous, retail, etc.)
- Advanced: Architecture design, optimization, deployment

9. Conclusion

Computer Vision has evolved from a theoretical academic curiosity of the 1950s into one of the most transformative technologies of the 21st century. This comprehensive analysis reveals that the field operates at the intersection of mathematics, neuroscience, engineering, and computer science, creating a powerful framework for teaching machines to perceive and understand the visual world.

10. REFERENCES

1. IBM Research. "What Is Computer Vision?" IBM Think. Retrieved from <https://www.ibm.com/think/topics/computer-vision>
2. AWS. "What is Computer Vision? - Image recognition AI/ML." Retrieved from <https://aws.amazon.com/what-is/computer-vision/>
3. Hyperscience. "What is Computer Vision?" Retrieved from <https://www.hyperscience.ai/resource/computer-vision/>
4. V7 Labs. "What Is Computer Vision? [Basic Tasks & Techniques]." Retrieved from <https://www.v7labs.com/blog/what-is-computer-vision>
5. OpenCV. "What is Computer Vision in 2025? A Beginners Guide." OpenCV Blog. Retrieved from <https://opencv.org/blog/what-is-computer-vision/>
6. IJRASET. "Computer Vision and Image Segmentation." International Journal for Research in Applied Science & Engineering Technology. Retrieved from <https://www.ijraset.com/research-paper/computer-vision-and-image-segmentation>
7. UPGRAD. "Basic CNN Architecture Explained: Key Layers and Workflow." Retrieved from <https://www.upgrad.com/blog/basic-cnn-architecture/>
8. VISO.AI. "Faster R-CNN: Breakthrough in Object Detection Tech." Retrieved from <https://viso.ai/deep-learning/faster-r-cnn-2/>
9. KNIME. "Convolutional Neural Networks & Computer Vision." KNIME Blog. Retrieved from <https://www.knime.com/blog/convolutional-neural-networks-computer-vision>
10. Ultralytics. "Vision AI History: From Edge Detection to YOLOv8." Retrieved from <https://www.ultralytics.com/blog/a-history-of-vision-models>
11. Kotwel. "Exploring the History & Revolution of Computer Vision." Retrieved from <https://kotwel.com/exploring-the-history-revolution-of-computer-vision/>
12. Crunch IS. "How Does Computer Vision Work? 6 Real-World Computer Vision Use Cases by Industry." Retrieved from <https://crunch-is.com/blog/how-does-computer-vision-work-real-world-computer-vision-use-cases-by-industry/>
13. Leewayhertz. "A guide to computer vision: Techniques, operational approaches, and future trends." Retrieved from <https://www.leewayhertz.com/computer-vision/>
14. DataMites. "Computer Vision: An Overview of Techniques and Applications." Retrieved from <https://datamites.com/blog/computer-vision-an-overview-of-techniques-and-applications/>
15. DataCamp. "What is Computer Vision? A Beginner Guide to Image Analysis." Retrieved from <https://www.datacamp.com/blog/what-is-computer-vision>
16. ENCORD. "10 Most Exciting Applications of Computer Vision in 2024." Retrieved from <https://encord.com/blog/use-cases-for-computer-vision/>
17. ENCORD. "Guide to Image Segmentation in Computer Vision." Retrieved from <https://encord.com/blog/image-segmentation-for-computer-vision-best-practice-guide/>
18. V7 Labs. "Image Processing: Techniques, Types, & Applications [2024]." Retrieved from <https://www.v7labs.com/blog/image-processing-guide>

19. V7 Labs. "YOLO Algorithm for Object Detection Explained [+Examples]." Retrieved from <https://www.v7labs.com/blog/yolo-object-detection>
20. Superannotate. "Introduction to computer vision: History and applications." Retrieved from <https://www.superannotate.com/blog/introduction-to-computer-vision>