Module (Course Syllabus) Catalogue

2024-2025

College/ Institute	**Erbil Polytechnic University**	
Department	Information System Engineering Techniques	
Module Name	Advanced Deep Learning	
Module Code	ADL	
Degree	**Technical Diploma** ☐ **Bachler** ☐ **High Diploma** ☐ **Master** ■ **PhD** ☐	
Semester	2nd	
Qualification	PhD	
Scientific Title	Asst. Prof.	
ECTS (Credits)	6	
Module type	**Prerequisite** ☐ **Core** ■ **Assist.** ☐	
Weekly hours		
Weekly hours (Theory)	**(3) hr Class**	**(30) Total hrs Workload**
Weekly hours (Tutorial)	**(0) hr Class**	**(0) Total hrs Workload**
Number of Weeks	**14**	
Lecturer (Theory)	Ismael Abdulrahman	
E-Mail & Mobile NO.	ismael.abdulrahman@epu.edu.iq	
Lecturer (Practical)		
E-Mail & Mobile NO.		
Websites	https://academicstaff.epu.edu.iq/faculty/ismael.abdulrahman	

Course Book

Course Description	This course provides an in-depth exploration of advanced deep learning techniques, including Convolutional Neural Networks (CNNs), their architectures, and applications. It covers well-known CNN models such as LeNet-5, AlexNet, VGG, Inception, ResNet, Vision Transformers (ViT), and hybrid architectures like ConvNeXt. Additionally, the course delves into transfer learning, object detection (YOLO, Faster R-CNN, SSD), and generative models (GANs, VAEs). Students will also learn about Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRUs) for handling sequential and time-series data. Advanced topics such as hybrid classical-quantum deep learning models will be introduced. With a strong focus on practical implementation, students will gain hands-on experience using Python frameworks like TensorFlow and PyTorch, applying deep learning to real-world problems in image recognition, medical diagnosis, sequence modeling, and generative AI.
Course objectives	By the end of this course, students will be able to: 1. Begin with fundamental gradient descent calculations, including numerical examples for optimization understanding. 2. Understand the fundamental concepts and history of advanced architectures of CNNs. 3. Explore well-known CNN architectures such as LeNet-5, AlexNet, VGG, Inception, ResNet, and other transformer-based techniques such as Vision Transformers (ViT) and hybrid models like ConvNeXt. 4. Apply transfer learning to improve model performance and reduce training time. 5. Implement object detection techniques using YOLO (versions 3, 5, and 8), Faster R-CNN, and SSD. 6. Gain hands-on experience in generative models, including Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs). 7. Develop deep learning models using TensorFlow and PyTorch. 8. Analyze and optimize deep learning models for performance improvements.

<table>
<tr><td></td><td colspan="4">9. Learn Python coding to implement deep learning solutions for CNN architectures, transfer learning, object detection, and generative models.

10. Understand and apply Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRUs) for time series forecasting and sequential data problems.

11. Explore advanced topics such as hybrid classical-quantum deep learning models for specialized tasks.</td></tr>
<tr><td>Student's obligation</td><td colspan="4">• Attendance
• Quizzes / homework / simulation / seminars/reports/ projects
• Exams (mid-term, final)
• Homework assignments will be a mix of paperwork and electronic copies. Written homework should be finished individually, discussions with peers or instructor are allowed. You will be given one week to finish the written homework. Some of the machine problems are designed for teamwork and due day may vary. Any late submission will not be considered.</td></tr>
<tr><td>Required Learning Materials</td><td colspan="4">Note:

Please be aware that assessment criteria may undergo alterations, including temporary adjustments to point distribution.</td></tr>
<tr><td rowspan="14">Evaluation</td><td>Task</td><td>Weight (Marks)</td><td>Due Week</td><td>Relevant Learning Outcome</td></tr>
<tr><td>Paper Review</td><td></td><td></td><td></td></tr>
<tr><td>Assignments: Homework</td><td></td><td></td><td></td></tr>
<tr><td>Assignments: Class Activity</td><td></td><td></td><td></td></tr>
<tr><td>Assignments: Report</td><td></td><td></td><td></td></tr>
<tr><td>Assignments: Seminar</td><td>5</td><td></td><td></td></tr>
<tr><td>Assignments: Essay</td><td>20</td><td></td><td></td></tr>
<tr><td>Assignments: Project</td><td></td><td></td><td></td></tr>
<tr><td>Quiz</td><td>5</td><td></td><td></td></tr>
<tr><td>Lab.</td><td>0</td><td></td><td></td></tr>
<tr><td>Midterm Exam</td><td>20</td><td></td><td></td></tr>
<tr><td>Final Exam</td><td>50</td><td></td><td></td></tr>
<tr><td>Total</td><td>100</td><td></td><td></td></tr>
</table>

Specific learning outcome:	Upon successful completion of this course, students will be able to: • Design and train CNN models for complex image classification tasks. • Compare and implement different CNN architectures, including LeNet-5, AlexNet, Inception, VGG, ResNet, Vision Transformers (ViT), and ConvNeXt. • Fine-tune pre-trained models for specific tasks using transfer learning. • Build object detection models using YOLO (versions 3, 5, 8), Faster R-CNN, and SSD for real-world applications. • Develop deep learning solutions for sequential and time-series problems using RNNs, LSTMs, and GRUs. • Generate new data using generative models like GANs and VAEs. • Understand and apply classical gradient descent techniques with numerical calculations. • Explore hybrid classical-quantum models for deep learning. • Optimize deep learning models using techniques such as batch normalization, dropout, and data augmentation. • Implement deep learning applications using Python frameworks like TensorFlow and PyTorch. • Deploy deep learning models for real-world practical applications.
Course References:	1. Elgendy, M. (2020). Deep Learning for Vision Systems. 2. Goodfellow etal. (2016). Deep Learning (Adaptive Computation and Machine Learning series). MIT Press. 3. Prince, S. J. D. (2024). Understanding Deep Learning. 4. Glassner, A. (2021). Deep Learning: A Visual Approach.

Course topics (Theory)	**Week**	**Learning Outcome**

Week 1: Review of Deep Learning Concepts

- Multilayer Perceptron (MLP):
  - Architecture and structure
  - Forward propagation and backpropagation
  - Gradient Descent (GD) and Stochastic Gradient Descent (SGD)
  - Activation functions (Sigmoid, ReLU)
  - Loss functions (MSE. RMSE, MAE, Cross-Entropy)
  - Training an MLP (Hyperparameters, Optimizers)

Week 2: Convolutional Neural Networks (CNN)

- Introduction to CNNs:

- Architecture: Convolution layers, Pooling layers, Fully connected layers
- Understanding Filters and Feature Maps
- Activation Functions and Batch Normalization
- Evaluation Metrics: Accuracy, Precision, Recall, F1-score

- CNNs for Image Recognition:
  - How CNNs are applied to image data
  - Advantages of CNNs over traditional neural networks
  - Project: Image classification with CNNs
- Training a CNN Model:
  - Using Transfer Learning (e.g., using pre-trained models like VGG, ResNet)

Week 3-5: CNN Architectures

- Week 3: Classic CNN Architectures
  - LeNet, AlexNet
  - Key ideas and innovations in early CNN architectures
  - Project: Image classification using classic architectures
- Week 4: Modern CNN Architectures
  - VGG, Inception, GoogLeNet, ResNet
  - Deep Residual Networks and skip connections
  - DenseNet and Inception Networks
  - Project: Image recognition with modern architectures
- Week 5: Advanced CNN Architectures
  - MobileNet, EfficientNet
  - CNNs for real-time applications and low-computation devices
  - Project: Object detection and segmentation with advanced CNNs

---

Week 6: Transfer Learning

- Introduction to Transfer Learning:
  - Why transfer learning works and when to use it
  - Pre-trained models (VGG, ResNet, Inception)
- Fine-Tuning Pre-trained Models:
  - Adjusting the last layers to your task
  - Freezing layers and fine-tuning
- Applications of Transfer Learning:
  - Image classification, NLP, time series forecasting

Week 7: Object Detection

- Introduction to Object Detection:
  - Overview of object detection tasks and use cases
  - Difference between image classification and object detection
- Popular Object Detection Models:
  - R-CNN, Fast R-CNN, Faster R-CNN
  - YOLO (You Only Look Once) and SSD (Single Shot Multibox Detector)
- Model Training for Object Detection:
  - Data preparation (annotating datasets)
  - Training an object detection model (Transfer learning in object detection)

Week 8: Generative Models

- Introduction to Generative Models:
  - Overview of generative vs discriminative models
  - Types of generative models (Autoencoders, GANs)
- Autoencoders:
  - Architecture and applications in dimensionality reduction
  - Variational Autoencoders (VAEs)
- Generative Adversarial Networks (GANs):
  - GAN architecture (Generator vs Discriminator)
  - Training GANs and their applications in generating new data

---

Week 9-10: Transformers and Hybrid Models

- Week 9: Introduction to Transformers
  - Attention Mechanisms and the Transformer architecture
  - Self-attention, Multi-head attention
  - Applications of Transformers in NLP (e.g., BERT, GPT)
- Week 10: Hybrid Models
  - Vision Transformers (ViT): Applying transformers to image data
  - ConvNeXt: Hybrid CNN-transformer architectures
  - Integrating CNNs with Transformers for improved performance in image and video processing

---

Week 11: Quantum Deep Learning

- Introduction to Quantum Computing:
  - Basic quantum principles (qubits, superposition, entanglement)
  - Quantum gates and circuits
- Quantum Neural Networks:
  - The concept of quantum deep learning
  - Quantum-inspired models (e.g., Quantum Convolutional Neural Networks)
  - Tools: PennyLane, Qiskit
- Applications of Quantum Deep Learning:
  - Quantum models for optimization, classification, and regression tasks

---

Week 12-14: Advanced Topics

- Week 12: Recurrent Neural Networks (RNN)
  - Structure and use cases for RNNs
  - Limitations of RNNs (vanishing gradient problem)
- Week 13: Long Short-Term Memory (LSTM)
  - LSTM architecture and working with sequential data
  - Use cases for time series prediction and text generation
- Week 14: Gated Recurrent Units (GRU)
  - GRU architecture and comparison to LSTMs
  - Applications in sequence modeling and time series forecasting

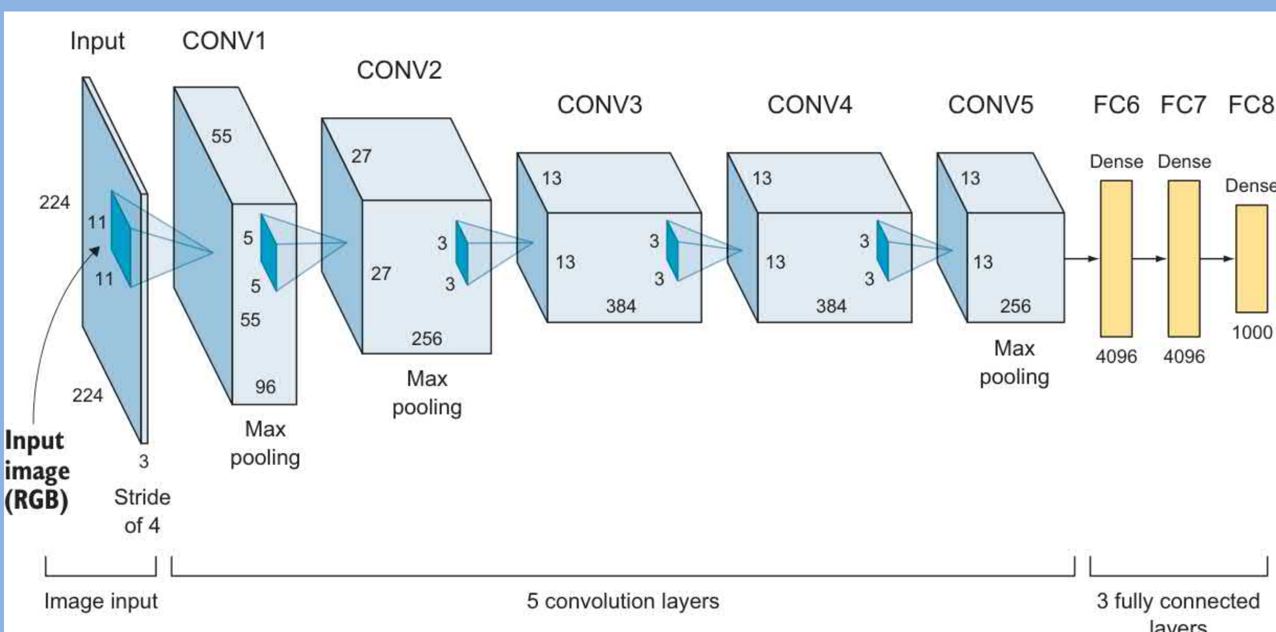
Note:

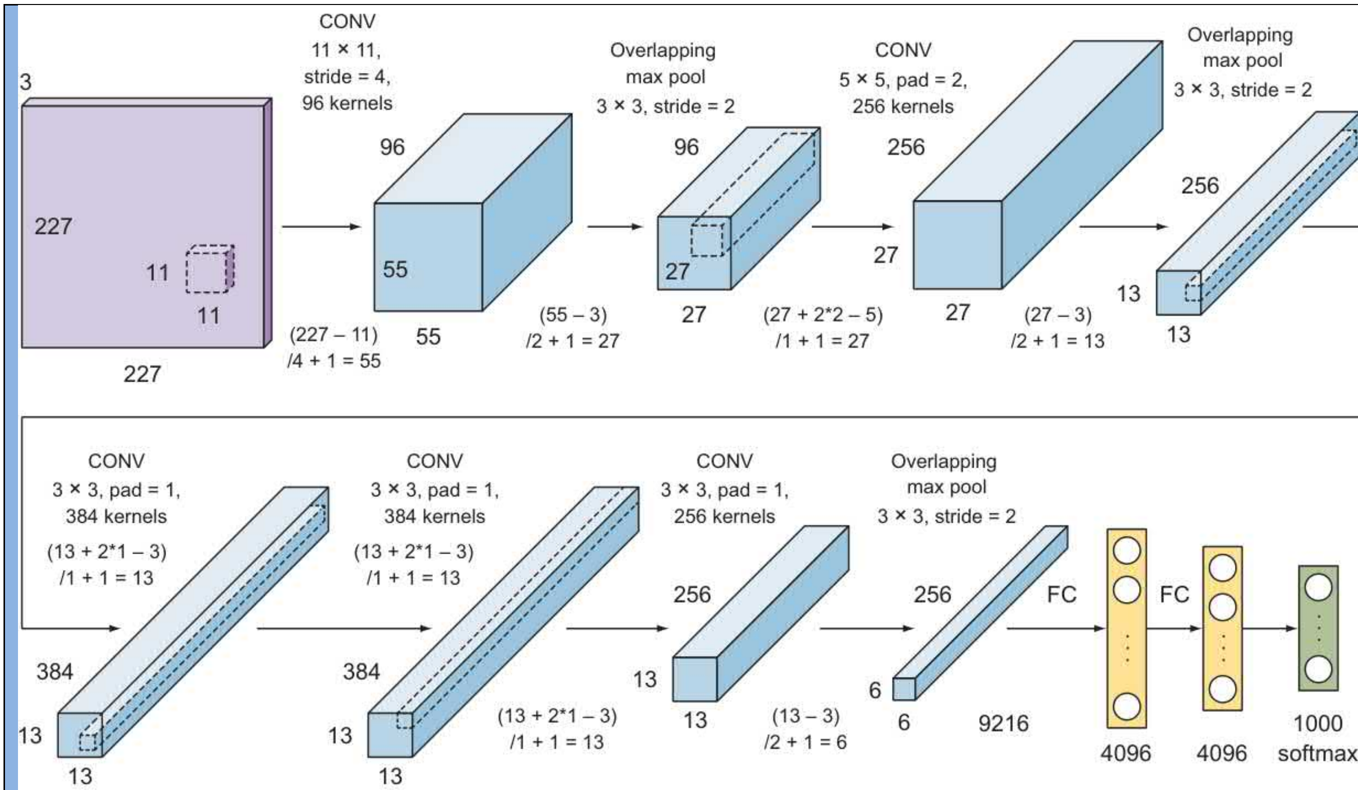
This syllabus serves as a flexible guide and may be adjusted based on the class's pace and students' depth of understanding. Practical demonstrations, guest lectures, and real-world applications will be incorporated to enhance the learning experience.

Practical Topics	Week	Learning Outcome
Building, Training, and Evaluating Deep Learning Models with Python, including CNNs, Object Detection, and other Architectures		

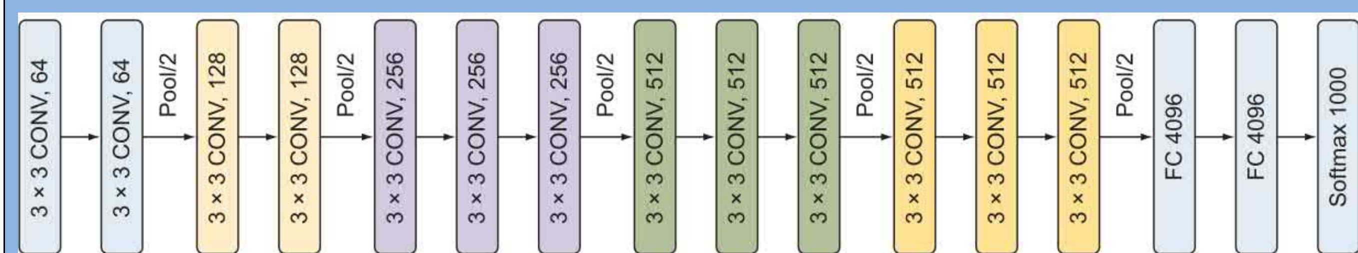
19. Examinations (samples of questions)

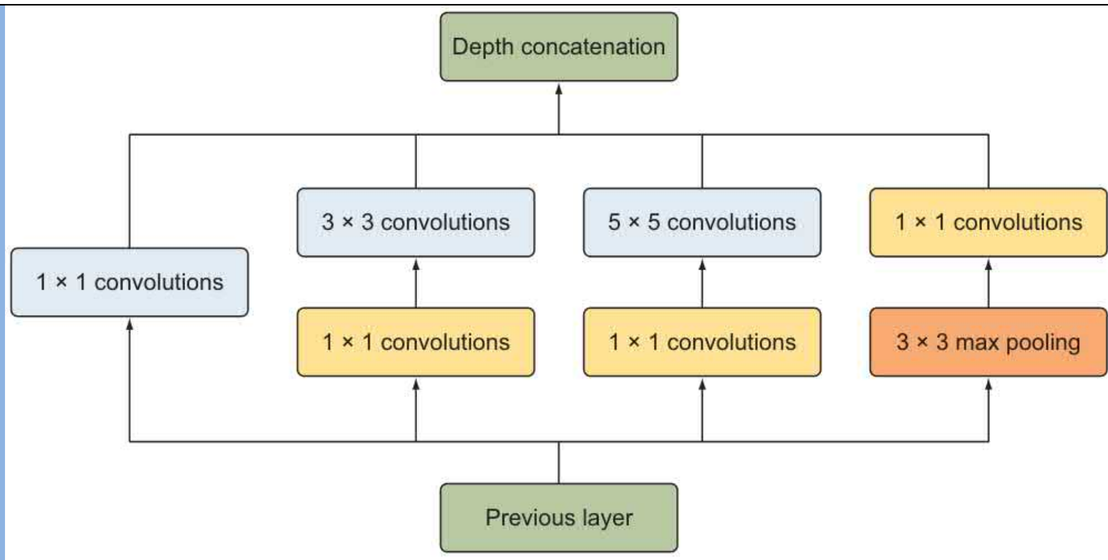
Q: Show the computations behind the numerical values of the parameters in the figures below. How many total parameters need to be trained and optimized? Show them in steps.

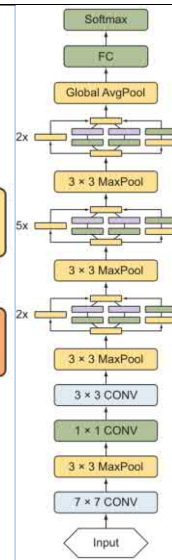




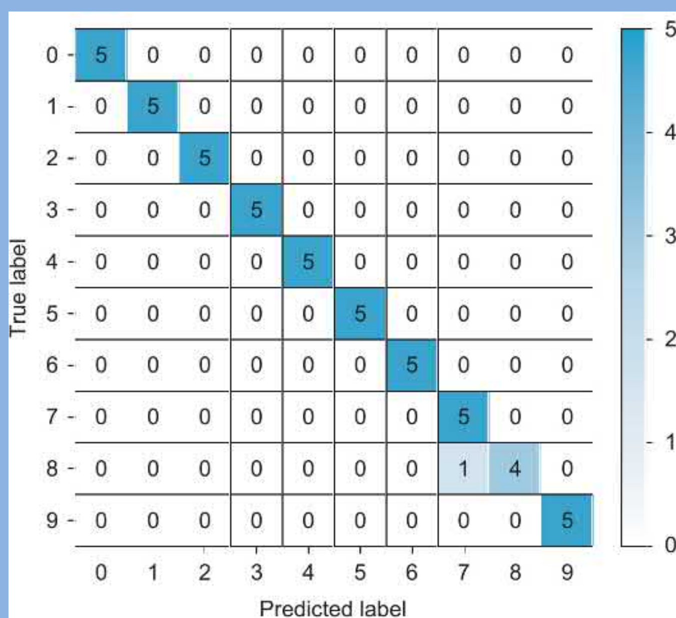
Q: What CNN architecture is shown in the figures below? Explain it in detail.



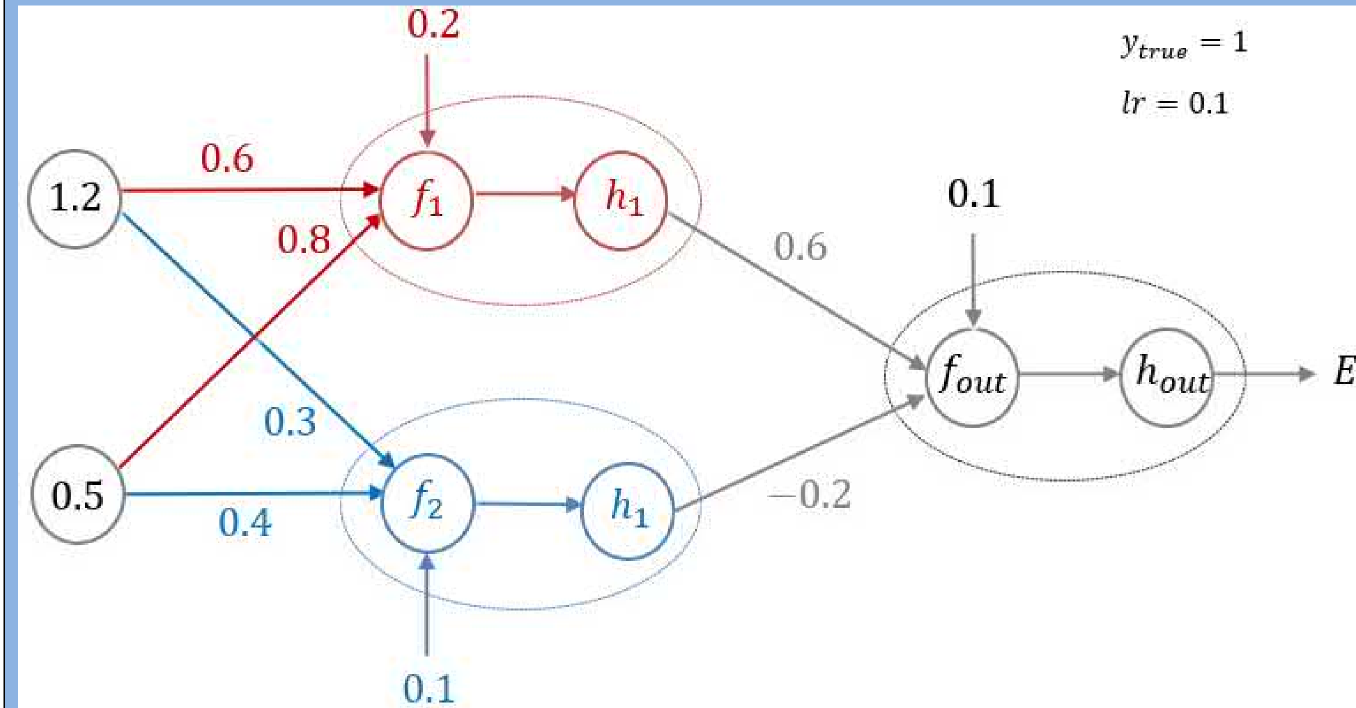




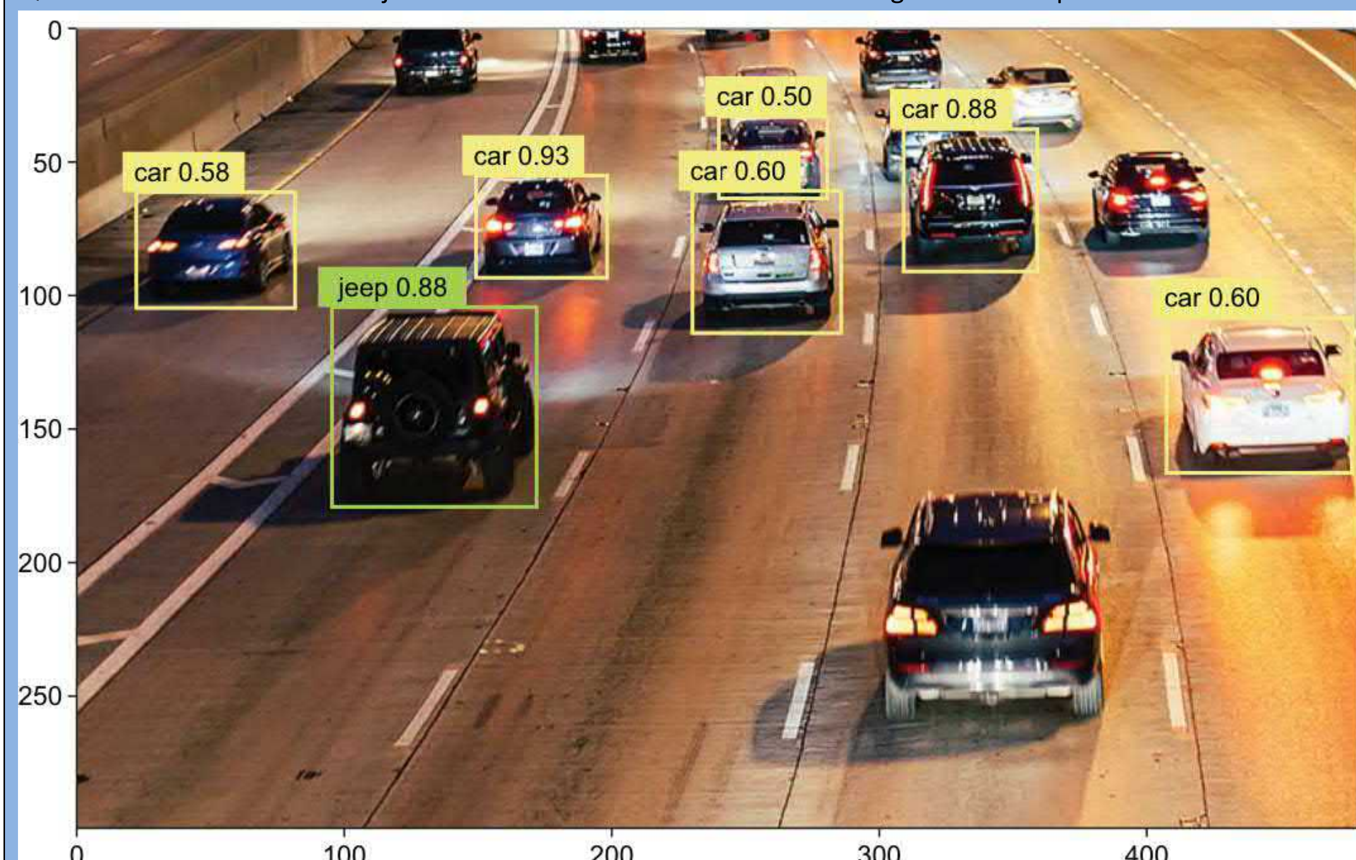
Q: What is this table for and how do you express the values?



Q: Compute the first epoch of gradient descent for the following network.



Q: Describe the methods of object detection. What do the values in the figure below represent?



Q: Describe the architecture below:

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 32, 32, 32)	896
activation_1 (Activation)	(None, 32, 32, 32)	0
batch_normalization_1 (batch	(None, 32, 32, 32)	128
conv2d_2 (Conv2D)	(None, 32, 32, 32)	9248
activation_2 (Activation)	(None, 32, 32, 32)	0
batch_normalization_2 (batch	(None, 32, 32, 32)	128
max_pooling2d_1 (MaxPooling2	(None, 16, 16, 32)	0
dropout_1 (Dropout)	(None, 16, 16, 32)	0
conv2d_3 (Conv2D)	(None, 16, 16, 64)	18496
activation_3 (Activation)	(None, 16, 16, 64)	0
batch_normalization_3 (batch	(None, 16, 16, 64)	256
conv2d_4 (Conv2D)	(None, 16, 16, 64)	36928
activation_4 (Activation)	(None, 16, 16, 64)	0
batch_normalization_4 (batch	(None, 16, 16, 64)	256
max_pooling2d_2 (MaxPooling2	(None, 8, 8, 64)	0
dropout_2 (Dropout)	(None, 8, 8, 64)	0
conv2d_5 (Conv2D)	(None, 8, 8, 128)	73856
activation_5 (Activation)	(None, 8, 8, 128)	0
batch_normalization_5 (batch	(None, 8, 8, 128)	512
conv2d_6 (Conv2D)	(None, 8, 8, 128)	147584
activation_6 (Activation)	(None, 8, 8, 128)	0
batch_normalization_6 (batch	(None, 8, 8, 128)	512
max_pooling2d_3 (MaxPooling2	(None, 4, 4, 128)	0
dropout_3 (Dropout)	(None, 4, 4, 128)	0
flatten_1 (Flatten)	(None, 2048	0
dense_1 (Dense)	(None, 10)	20490

External supporting Link:

YouTube lectures are prepared and recorded by the instructor:

https://www.youtube.com/@ismaelabdulrahman7295/videos

External Evaluator

I confirm that the syllabus given the attached course book is sufficient and covers the required areas needed for the students.