| CSE----Computer Vision and Pattern Recognition | | |
|---|---|---|
| Teaching Scheme | Examination Scheme | Credits allocated |
| Theory 4 h/week | End of semester Examination-60 marks | Theory-4 |
| **Course Prerequisite: Students should have knowledge of Liner Algebra, Data Structures and Programming** | | |
| **Course Objective:**<br>**1.** To understand concepts of Digital Image<br>**2.** To understand the implementations Image vision applications | | |
| **Course Outcomes:** On completion this course, students will be able to | | |
| 1. Toapplymathematical modeling methods for low, intermediate and high- level image pro- cessing tasks<br>2. To be able to design new algorithms to solve recent state of the art computer vision problems.<br>3. To perform software experiments on computer vision problems and compare their performance with the state of the art.<br>4. To build a complete system to solve a computer vision problem. | | |
| **Level** | Masters | |
| **Course Content:** | | |
| Unit –I | **Introduction to Computer Vision**<br>Definition and applications of computer vision, Historical background and development, Importance and relevance in modern technology. | 10 hrs |
| Unit-II | **Image Basics and Image processing Techniques**<br>Understanding digital images, Image representation in computers,Image enhancement: histogram equalization, contrast stretching, Image operations (blurring, sharpening, edge detection), Filtering techniques: smoothing, sharpening, edge detection, Morphological operations: erosion, dilation, opening, and closing. | 10 hrs |
| Unit-III | **Image Classification and Segmentation**<br>Feature Extraction: Introduction to feature extraction techniques, Feature descriptors: Histogram of Oriented Gradients (HOG), ScaleInvariant Feature, Image Classification: SVM, Decision Trees, Gradient Boosting Machines, Naïve Bayes, Image Segmentation: Thresholding, Region-based segmentation, Edgebased segmentation, Semantic segmentation, Instance segmentation. | 10 hrs |
| Unit-IV | **Convolutional Neural Networks**<br>Basic architecture of CNNs: Convolutional layers, pooling layers, Activation functions, Tools and Libraries: Introduction to deep learning frameworks, Hands-on coding exercises using high-level APIs (Keras), Training CNNs-Dataset preparation and preprocessing, Loss functions (cross- entropy), Optimizers (e.g., SGD, Adam), Back-propagation in CNNs, Applications of CNNs in Computer Vision-Image classification, Object detection, Semantic segmentation. | 10 hrs |
| | | |
| **Internal assessment** | | |
| **Part A** | CIA-I: Unit I, and II | 20 Marks |
| | CIA-II: Unit III, and IV | 20 Marks |

| Part B | ESE: Term Exam | 60 Marks |
|---|---|---|

**Text/Reference Books:**


1. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag London Limited 2011.
2. Computer Vision: A Modern Approach, D. A. Forsyth, J. Ponce, Pearson Education, 2003.
3. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press, March 2004.
4. Christopher M. Bishop; Pattern Recognition and Machine Learning, Springer, 2006
5. R.C. Gonzalez and R.E. Woods, Digital Image Processing, Addison- Wesley, 1992.

# CSE464: System Software

**B.Tech. (CSE) Elective course**

**Credits 04 (3L+1T+0P)**

**Learning Objectives:**

1. To understand the relationship between system software and machine architecture.
2. To understand the processing of an HLL program for execution on a computer.
3. To understand the process of Assembling a program
4. To understand the process of Loading and Linking a program
5. To understand the process of Compiling a program.

**Unit I:**

**Background:** System software and Machine Architectures, The Simplifies Instructional Computer (SIC & SIC/XE), Traditional CISC machines, RISC machines.

**Assemblers:**Basic Assembler functions, Machine- Dependent Assembler features, machine-Independent Assembler features, Assembler design options, Implementation examples.

**Unit II:**

**Loaders & Linkers:**Basic loader functions, Machine-dependent Loader features, machine-independent Loader features, Loader design options, Implementation examples.

**Unit III:**

**Macro Processors:** Basic Macro Processor functions, Machine-Independent Macro Processor features, Macro processor design options, implementation examples.

**Unit IV:**

**Compilers:** Basic Compiler functions, Machine-dependent Compiler features, Machine-independent Compiler features, Compiler design options, Implementation examples.

**Reference books:**

1. Leland L. Beck, “System Software – An Introduction to Systems Programming”, 3rd Edition, Pearson Education Asia, 2000.

2. SantanuChattopadhyay, “System Software”, Prentice-Hall India, 2007.

3. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, “Compilers: Principles, Techniques, and Tools”,2nd Edition, Pearson Education Asia.