# COURSE DESCRIPTION CARD - SYLLABUS

Course name
Computer vision [S1SI1E>WKOM]

## Course

Field of study
Artificial Intelligence

Year/Semester
3/5

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
English

Form of study
full-time

Requirements
elective

## Number of hours

Lecture
30

Laboratory classes
30

Other
0

Tutorials
0

Projects/seminars
0

## Number of credit points

5,00

## Coordinators

prof. dr hab. inż. Krzysztof Krawiec
krzysztof.krawiec@put.poznan.pl

dr inż. Krzysztof Martyn
krzysztof.martyn@put.poznan.pl

## Lecturers

## Prerequisites

Basic mathematical knowledge of mathematical analysis and linear algebra. Programming skills in Python (preferably including the familiarity with the Numpy library). Knowledge acquired during the courses on Introduction to AI, Statistics, and Machine learning.

## Course objective

1. To make the students familiar with the fundamentals of computer vision and image processing, with emphasis on connections with artificial intelligence and computational intelligence, and in particular with machine learning. 2. To help the students in developing skills of solving problems pertaining to image processing, object detection, segmentation, classification and localization, and acquiring experience in applying those skills in selected practical applications. 3. To help the students in developing capabilities of leading and participating in small software projects involving image processing and analysis, including teamwork skills and good practices.

## Course-related learning outcomes

Knowledge:
The student has a structured and theoretically founded general knowledge related to key issues in the field of computer vision [K2st_W2].
Has advanced detailed knowledge regarding selected topics in computer vision, in particular image acquisition and image processing, scene analysis, and design of systems that perform scene-based reasoning [K2st_W3].
Has advanced and detailed knowledge of the processes occurring in the life cycle of computer vision systems, including data acquisition techniques, and designing, testing and deployment of such systems [K2st_W5].
Knows advanced methods, techniques and tools used to solve complex engineering tasks and conduct research within computer vision, in particular the methodology pertaining to conducting computational experiments and metrics for assessment of the quality of computer vision systems [K2st_W6].

Skills:
The student is able to plan and carry out experiments, including computer measurements and simulations, interpret the obtained results and draw conclusions and formulate and verify hypotheses related to complex engineering problems and simple research problems in computer vision [K2st_U3].
Can use analytical, simulation and experimental methods to formulate and solve engineering problems and simple research problems in computer vision [K2st_U4].
Can - when formulating and solving engineering tasks - integrate knowledge from different areas of computer science (and if necessary also knowledge from other scientific disciplines) and apply a systemic approach, also taking into account non-technical aspects [K2st_U5].
Is able to assess the suitability and the possibility of using new achievements (methods and tools) and new IT products [K2st_U6].
Can carry out a critical analysis of existing technical solutions used in computer vision systems and propose their improvements (streamlines) [K2st_U8].
Is able - using among others conceptually new methods - to solve complex tasks involving design and implementation of computer vision systems, including atypical tasks and tasks containing a research component [K2st_U10].
Is able - in accordance with a given specification, taking into account non-technical aspects - to design a complex device, IT system or process typical for the computer vision area and implement this project - at least in part - using appropriate methods, techniques and tools, including adapting to this purpose existing tools or developing new ones [K2st_U11].

Social competences:
Social competencies
The student understands that in the field of computer vision the knowledge and skills quickly become obsolete [K2st_K1].
Understands the importance of using the latest knowledge in the field of computer vision in solving research and practical problems [K2st_K2].

## Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Formative assessment:
1. lectures
a. asking student questions pertaining to the material presented in previous lectures,
2. laboratory classes:
a. evaluation of progress in project realization (checkpointing)
b. checking knowledge and skills through homeworks
Total assessment:
1. verification of assumed learning objectives related to lectures:
a. Evaluation of acquired knowledge in the form of a written exam (5-8 open questions pertaining to lecture contents). Roughly half of questions are theoretical (define, describe, characterize, etc.), the other half are practical and require manual calculations (e.g., apply the erosion algorithm to a small binary image). Maximum total score: 25 points, of which 13 are required to obtain a positive grade.

2. verification of assumed learning objectives related to laboratory classes:
a. After the first 6 classes, students solve practical programming assignments and report their solutions to the instructors leading the laboratory classes within one week. Each assignment is evaluated on a scale from 2.0 to 5.0.
b. Evaluation of progress along the semester classes, based on 3 projects carried out by students and based on its documentation; students work on the project in part during the classes, and partially individually. The assigned grade reflects also student's teamwork skills.
c. The final grade is computed as a weighted average from the individual marks. Homeworks constitute 40% of the final grade and each projects 20%.
Additional assessment elements include:
● Student's capability of applying the acquired knowledge to the problem posed in the project.
● Student's remarks aimed at improving the quality of teaching material.
● Indications of students' problems at acquisition and understanding of the knowledge presented at the lectures, aimed at improving the overall quality of the teaching process.

## Programme content

Lecture:
Introduction. The place of computer vision within computer science, artificial intelligence, and related disciplines. Application areas and scientific reading. Characteristics of visual information. Classes of images and methods of image representation. Image parameters: dimensions, depth, resolution. Color representation spaces.
Single-point image processing. Gamma correction. Pseudocoloring. Image arithmetic. Linear filtering. Definition and properties of convolution. Frequency-domain image processing. Fourier transform. Interpretation of image spectrum. Morphological image processing: dilation, erosion, opening, closing. Effective algorithms for morphological filtering. Generalization to grayscale images. Discrete geometry. Edge detection and contour following algorithms. Hough transform.
Shape and texture features. Measurement of object dimensions and shape. Shape coefficients. Skeletons. Geometrical moments. Fractal dimension. Textural features: autocorrelation, co-occurrence matrix, spectral representations. Structural methods for texture description. Image segmentation. Classes of algorithms for image segmentation: thresholding, edge detection, region growing and region splitting. Distance transform. Watershed segmentation.
Fundamentals of image acquisition and Imaging sensors. Perspective camera model and other camera models. External and internal camera parameters. Stereovision. Canonical stereo camera system (simple stereo). The concept of disparity. Methods of depth estimation from disparity. Other depth estimation methods, including methods based on a single view and methods using machine learning.

Lab:
Laboratory classes are conducted in the form of fifteen 2-hour exercises, held in the laboratory, preceded by a 6-hour instructional session at the beginning of the semester. The exercises are carried out by teams of 2 students. The laboratory program includes the following topics: Introduction (2h): Presentation of the assumptions of the laboratory part of the subject. Presentation of computer tools used in the laboratory part (programming libraries, programming environments). Instructional session (6h): Exercises involving the implementation of selected methods of image processing and analysis in popular programming languages (Python, C++). Testing of implemented algorithms on real and artificial images. Evaluation of the correctness and effectiveness of the algorithms. Good practices for the design and implementation of algorithms for image processing and analysis. Typical errors and ways to avoid them. Implementation of projects in groups (22h): Realization, in groups of two,
of programming projects aimed at programmatic implementation of specific functionalities of image processing and analysis.
The classes are divided into three sections: (1) image representation and processing, (2) image description, and (3) neural networks. The first part covers issues related to single point processing, image arithmetic, geometric transformations, convolution, edge detection, blur filters, nonlinear filters, morphological operations, frequency domain processing and Fourier transform. The second section includes key point detection, pattern recognition, higher-level pattern building, point, edge, area and semantic segmentation, detection of geometric features, sample algorithms implemented in the OpenCV library, the principle of operation of the camera and video processing. The last part concerns neural

networks and their architectures used in computer vision.

## Course topics

none

## Teaching methods

1. Lectures: multimedia presentation, illustrated with examples, with occasional use of black board. Software demonstration.
2. Labs: practical exercises, problem solving, design and implementation of image analysis systems, performing computational experiments, discussion, teamwork, presentation of project outcomes (software and computational experiments).

## Bibliography


Basic:
1. Gonzalez, R.C, Wintz, P., Digital Image Processing. Addison-Wesley 2017
2. Szeliski, R. Computer vision: algorithms and applications. Springer Nature, 2022. Available for free for personal use at https://szeliski.org/Book/
3. Goodfellow, I., Bengio, Y., Courville, A., Deep learning. MIT Press, 2016. Available online for free at https://www.deeplearningbook.org/

Additional:
1. Zieliński, T.P., Cyfrowe przetwarzanie sygnałów. WKŁ 2009.
2. Cyganek, B., Komputerowe przetwarzanie obrazów trójwymiarowych. EXIT 2002.
3. Owen, M., Przetwarzanie sygnałów w praktyce. WKŁ 2009.
4. Choraś, R. Komputerowa wizja. Metody interpretacji i identyfikacji obiektów. EXIT, 2006.


## Breakdown of average student's workload

| | Hours | ECTS |
|---|---|---|
| Total workload | 125 | 5,00 |
| Classes requiring direct contact with the teacher | 62 | 2,50 |
| Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation) | 63 | 2,50 |