

# Lehrveranstaltung

## IBA - Industrielle Bildanalyse

---

Version: 4 | Letzte Änderung: 23.09.2019 09:14 | Entwurf: 0 | Status: vom verantwortlichen Dozent freigegeben

### ^ Allgemeine Informationen

<b>Langname</b>	Industrielle Bildanalyse
<b>Anerkennende LModule</b>	<a href="#">IBA_BaET</a> , <a href="#">BV2_BaMT</a> , <a href="#">IBA_BaTIN</a>
<b>Verantwortlich</b>	Prof. Dr. Lothar Thieling Professor Fakultät IME
<b>Niveau</b>	Bachelor
<b>Semester im Jahr</b>	Wintersemester
<b>Dauer</b>	Semester
<b>Stunden im Selbststudium</b>	78
<b>ECTS</b>	5
<b>Dozenten</b>	Prof. Dr. Lothar Thieling Professor Fakultät IME
<b>Voraussetzungen</b>	Grundlagen der Signalverarbeitung Grundlagen der Programmierung in Java oder C Grundlagen der Analysis und Linearen Algebra
<b>Unterrichtssprache</b>	deutsch
<b>separate Abschlussprüfung</b>	Ja

## Abschlussprüfung

### Details

Die Studierenden sollen in einer mündlichen Prüfung folgende Kompetenzen nachweisen: 1.) Sicherer Umgang mit grundlegenden Begrifflichkeiten, Mechanismen und Konzepten. 2.) Problemstellungen aus dem Bereich der Industriellen Bildanalyse analysieren und mit passenden Methoden lösen können. 3.) Vorliegende Problemlösungen analysieren und die dabei verwendeten algorithmischen und theoretischen Grundlagen erklären können.

### Mindeststandard

Mindestens 50% der möglichen Gesamtpunktzahl.

## Prüfungstyp

Die Studierenden sollen in einer mündlichen Prüfung folgende Kompetenzen nachweisen: 1.) Sicherer Umgang mit grundlegenden Begrifflichkeiten, Mechanismen und Konzepten. 2.) Problemstellungen aus dem Bereich der Industriellen Bildanalyse analysieren und mit passenden Methoden lösen können. 3.) Vorliegende Problemlösungen analysieren und die dabei verwendeten algorithmischen und theoretischen Grundlagen erklären können.

## ^ Vorlesung / Übungen

### Lernziele

---

#### Kenntnisse

Bildaufbau und Zugriff auf Bilddaten

Bildmatrix

Grauwert- und Farbbilder

Entwicklungsumgebung

Software-Entwicklungsumgebung

Compiler

Linker

Debugger

Entwicklungsumgebung für die Bildverarbeitung und Bildanalyse

programmtechnischer Zugriff auf Bilddaten und Parameter

Überblick über die zur Verfügung stehenden BV-Module

Erstellung eigener BV-Module

Erstellung von "Algorithmenketten" auf Basis von BV-Modulen mittels grafischer Programmierung

---

Segmentierung

Histogrammbasierte Segmentierung

Histogrammanalyse

Shading und dessen Beseitigung

flächenbasierte Segmentierung

Filling

Split and Merge

Region Growing

kantenbasierte Segmentierung

Konturverfolgung

Hough-Transformation

---

Merkmalsextraktion

geometrische Merkmale

grundlegende Merkmale (Fläche, Umfang, Formfaktor)

Zentralmomente

normierte Zentralmomente

Polarabstand

Krümmungsverlauf

DFT von Polarabstand und/oder Krümmungsverlauf

Farbmerkmale (HSI)

Texturmerkmale

Klassifikation von Merkmalen

Begriffe und Grundlagen

Merkmalsvektor, Merkmalsraum, Objektklassen ...

überwachte/unüberwachte Klassifikation

lernende/nicht lernende Klassifikation

"klassische" Verfahren

Quadermethode

Minimum-Distance

Nearest Neighbour

Maximum-Likelihood

neuronale Netze

das künstliche Neuron als einfachster Klassifikator

Arbeitsweise

Aufgabe der Aktivierungsfunktion

Aufgabe des Bias

Training eines Neurons (Gradientenabstiegsverfahren)

Multi-Layer-Perceptron

Aufbau

Aufgabe der Layer

Backpropagation-Trainingsalgorithmus

Entwicklungsumgebung zur Erstellung und zum Training Neuronaler Netze

Erstellen und konfiguration neuronaler Netze

Training neuronaler Netze

Verifikation trainierter Netze

Erzeugung von C-Funktionen aus trainierten Netzen

---

## Fertigkeiten

die vorgestellten Verfahren zur Segmentierung

angeben

beschreiben

hinsichtlich der Einsatzfelder abgrenzen

hinsichtlich der Vor- und Nachteile bewerten

problemspezifisch parametrieren

---

die vorgestellten Merkmale und Verfahren zur Merkmalsextraktion

angeben

beschreiben

hinsichtlich der Einsatzfelder abgrenzen

hinsichtlich der Vor- und Nachteile bewerten

problemspezifisch parametrieren

---

die vorgestellten Verfahren zur Klassifikation

angeben

beschreiben

hinsichtlich der Einsatzfelder abgrenzen

hinsichtlich der Vor- und Nachteile bewerten

problemspezifisch parametrieren

# Aufwand Präsenzlehre

Typ	Präsenzzeit (h/Wo.)
Vorlesung	2
Übungen (ganzer Kurs)	0
Übungen (geteilter Kurs)	0
Tutorium (freiwillig)	0

## Separate Prüfung

keine

## ^ Praktikum

### Lernziele

---

#### Fertigkeiten

siehe Fertigkeiten, die unter "Vorlesung/Übung->Lernziele->Fertigkeiten" aufgeführt sind

---

komplexere Aufgaben in einem Kleinteam bewältigen

---

Erarbeitung von komplexeren Problemlösungen die sich mittels Bildverarbeitung und Bildanalyse implementieren lassen

komplexere Problemstellungen verstehen und analysieren

Systemverhalten aus spezifizierenden Texten herleiten

System strukturiert analysieren

sinnvolle Teilsysteme erkennen

Schnittstellen zwischen Teilsystemen erfassen

Gesamtsystem auf Basis von Teilsystemen modellieren

Auswahl geeigneter bekannter Verfahren

Modifikation bekannter Verfahren

Kombination geeigneter Verfahren

Teilsysteme modellieren, implementieren, testen

Teilsysteme soweit möglich auf zur Verfügung stehende Komponenten (BV-Module) abbilden, d.h. Modulauswahl und Parametrierung.

Nicht zur Verfügung stehende aber benötigte BV-Module mittels Software-Entwicklungsumgebung in C implementieren und testen

Compilieren (Finden syntaktischer Fehler und deren Behebung)

Debuggen (Finden semantischer Fehler und deren Behebung)

Gesamtsystem (Problemlösung) implementieren testen und validieren

Erstellung der Problemlösung als "Algorithmenkette" auf Basis von BV-Modulen mittels grafischer Programmierung

Parametrierung der BV-Module

Validierung der Problemlösung

Auf Basis der Validierungsergebnisse in Iterationszyklen die Algorithmenkette und die Parametrierung der BV-Module anpassen. Bei Bedarf auch die BV-Module selbst modifizieren.

## Aufwand Präsenzlehre

Typ	Präsenzzeit (h/Wo.)
Praktikum	2
Tutorium (freiwillig)	0

## Separate Prüfung

keine