


# EMOTIONSERKENNUNG MIT NEURONALEN NETZEN

Christian Rieker, Stuart Church

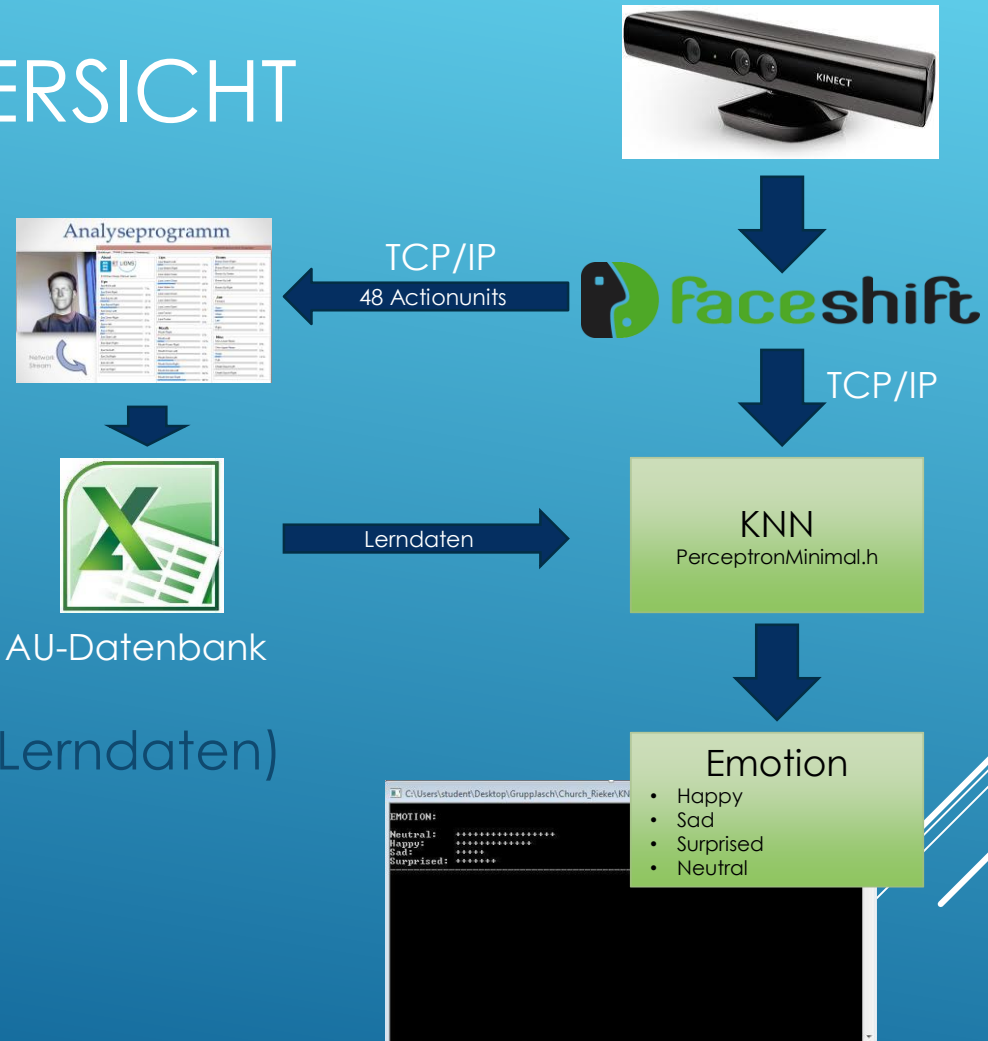
A decorative graphic consisting of several parallel white lines of varying thicknesses, slanted diagonally from the bottom left towards the top right, located in the lower right quadrant of the slide.

# INHALT

- ▶ Ansatz und Übersicht
  - ▶ Neuronales Netz
  - ▶ Lerndaten
  - ▶ Lerndaten erstellen
  - ▶ Optimierung
  - ▶ Neuronales Netz einlernen
  - ▶ Ergebnisse
  - ▶ Fazit & Ausblick
  - ▶ Demo
- 

# ANSATZ UND ÜBERSICHT

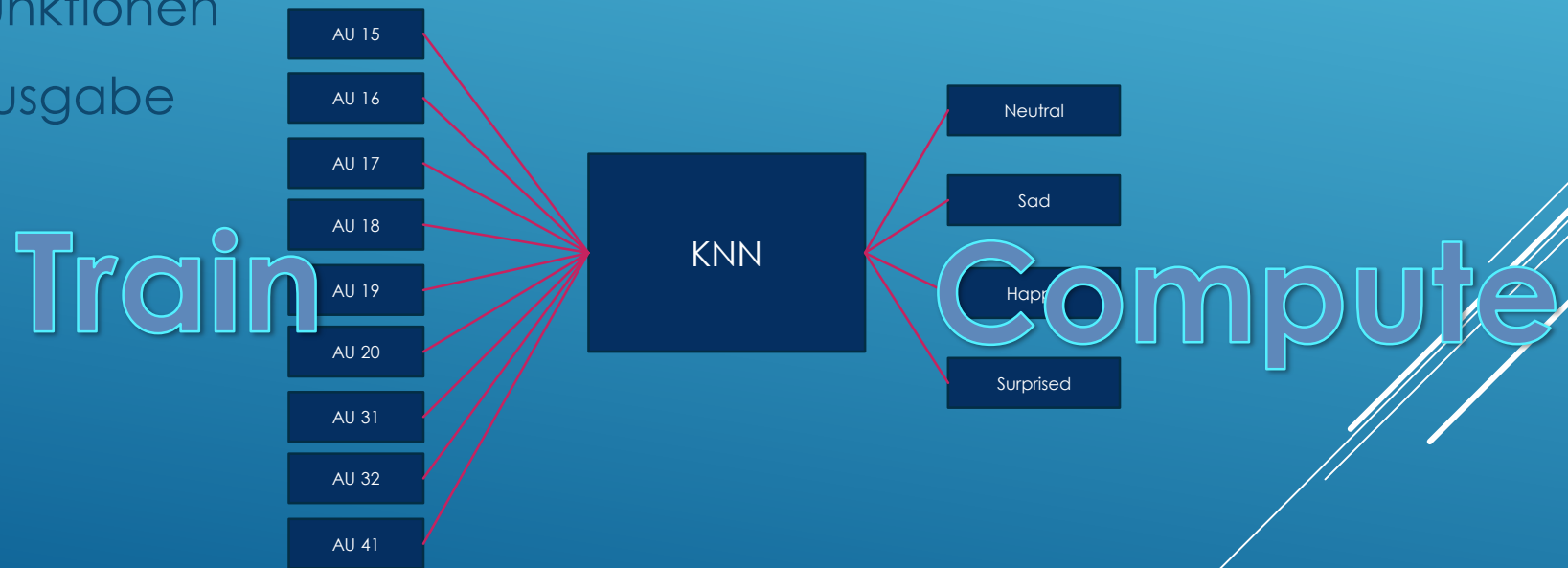
- ▶ Microsoft Kinect
- ▶ Faceshift
- ▶ Analyseprogramm
- ▶ Actionunits-Datenbank (Lerndaten)
- ▶ Ausgabe der Emotionen



# NEURONALES NETZ

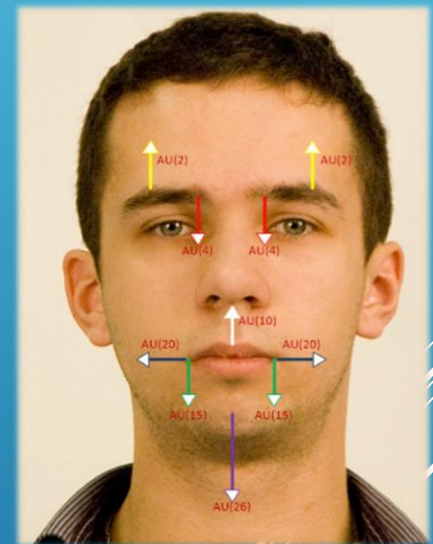
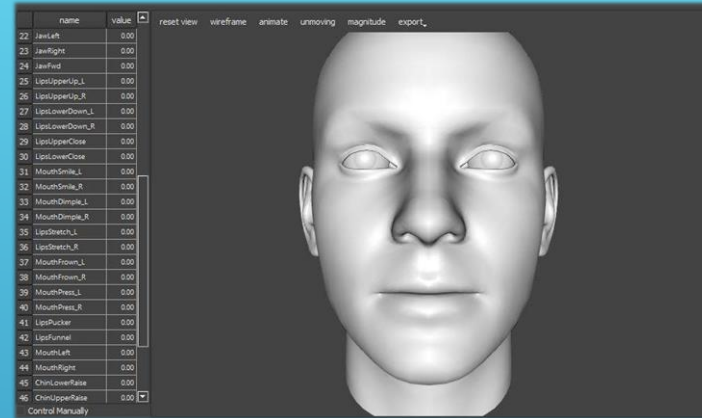


- ▶ C-Bibliothek PerceptronMinimal.h
- ▶ Bedeutung und Einstellung der Parameter
- ▶ Funktionen
- ▶ Ausgabe



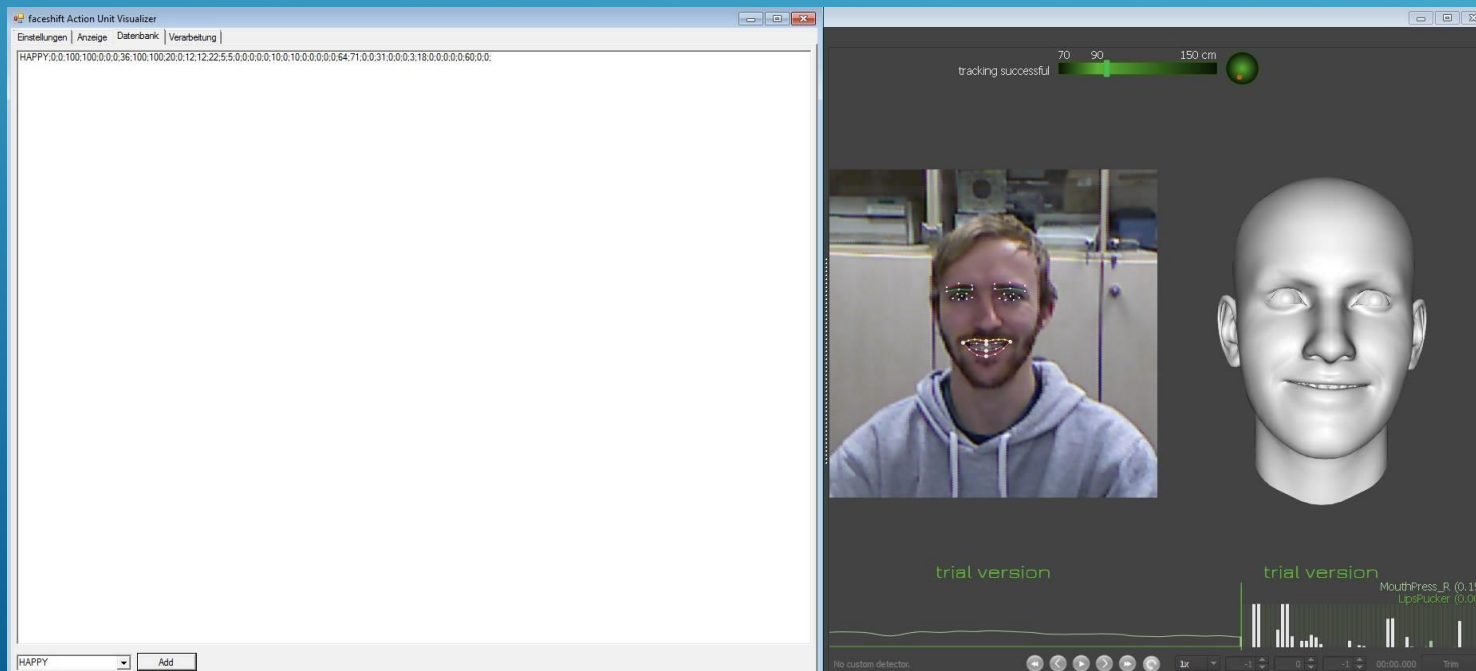
# LERNDATEN

- ▶ ActionUnits aus Faceshift
- ▶ Aufnahme der Daten mit Analyseprogramm
- ▶ Lernerfolg abhängig von Lerndaten
- ▶ Einlesen der Daten aus Exceltabelle
- ▶ Abfrage im KNN-Programm wie viele Lerndaten verwendet werden sollen
- ▶ Lerndaten müssen voneinander unabhängig sein



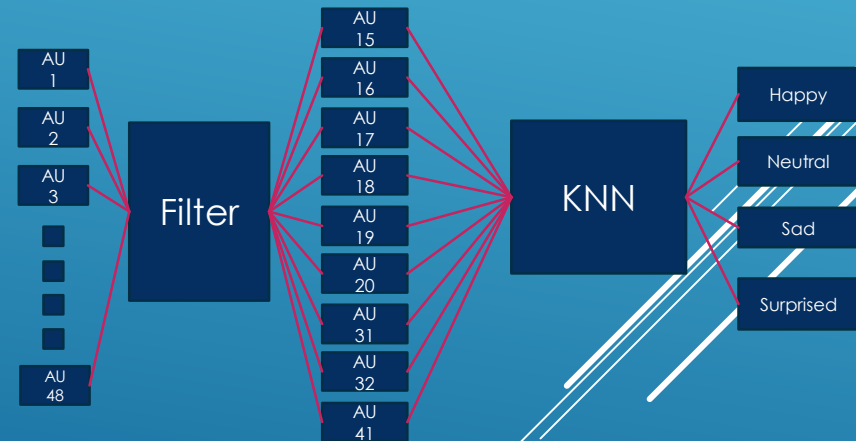
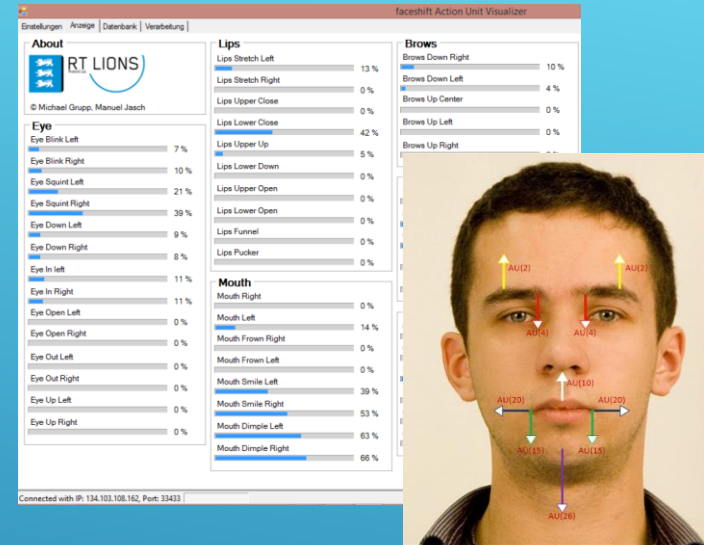
# LERNDATEN ERSTELLEN

- ▶ Verbindung Faceshift – Analyseprogramm via TCP/IP
- ▶ Emotion auswählen
- ▶ Actionunits werden in Excelfile abgespeichert



# OPTIMIERUNG

- ▶ Rechenzeit abhängig von Anzahl der Daten
- ▶ Viele Actionunits keinen Einfluss auf Emotion (Augenposition)
- ▶ Minimierung der Lerndaten auf 9 relevante Actionunits
- ▶ Minimierung der Eingangsdaten von Faceshift
- ▶ Festlegung der Ausgangsemotionen auf Neutral, Happy, Sad und Surprised
- ▶ Optimierung der Parameter des KNN



# NEURONALES NETZ EINLERNEN

- ▶ Benutzerinteraktion über Konsole
- ▶ Eingabe der Anzahl an Lerndaten die verwendet werden sollen
- ▶ Aus jeder Excelfile wird die entsprechende Anzahl Eingangsdaten gelesen
- ▶ Anzahl erwarteter Ausgangsdaten wird entsprechend angepasst



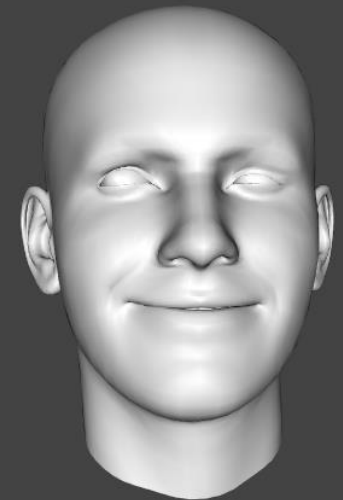
```
C:\Users\student\Desktop\Grupplasch\Church_Rieker\KNN\KNN\faceshift-network-parser\faceshif...
Please enter learn-data number of entries per emotion: 5
20
Training Cycle: 2000 Error= 6.35324
Training Cycle: 4000 Error= 6.01394
Training Cycle: 6000 Error= 5.66536
Training Cycle: 8000 Error= 5.45129
Training Cycle: 10000 Error= 5.25776
Training Cycle: 12000 Error= 5.08381
Training Cycle: 14000 Error= 4.92057
Training Cycle: 16000 Error= 4.76897
Training Cycle: 18000 Error= 4.61880
Training Cycle: 20000 Error= 4.48054
Training Cycle: 22000 Error= 4.33969
Training Cycle: 24000 Error= 4.20304
Training Cycle: 26000 Error= 4.07115
Training Cycle: 28000 Error= 3.93977
Training Cycle: 30000 Error= 3.82226
Training Cycle: 32000 Error= 3.69048
Training Cycle: 34000 Error= 3.55962
Training Cycle: 36000 Error= 3.44377
Training Cycle: 38000 Error= 3.30912
Training Cycle: 40000 Error= 3.18177
Training Cycle: 42000 Error= 3.07669
Training Cycle: 44000 Error= 2.95398
Training Cycle: 46000 Error= 2.85678
Training Cycle: 48000 Error= 2.74196
Training Cycle: 50000 Error= 2.65202
Training Cycle: 52000 Error= 2.53782
```




# ERGEBNISSE

- ▶ Lernergebnis stark abhängig von Anzahl und Güte Lerndaten
- ▶ Steuerung des Lernvorgangs über Parameter
- ▶ Gute Erkennung von Surprised und Happy

```
C:\Users\student\Desktop\Grupp\asch\Church_Rieker\KNN\KNN\faceshift-network-parser\faceshif...  
EMOTION:  
Happy:  ++++++ 76%  
Neutral: +      1%  
Surprised: ++++++ 10%  
Sad:    ++++++ 15%
```



# FAZIT & AUSBLICK

- ▶ Schwer zu beherrschender Vorgang
  - ▶ Eingangsdaten von Faceshift stark variabel
  - ▶ Einlernen mit verschiedenen Lerndaten hat Einfluss auf Lernergebnis
  - ▶ Lange Rechenzeiten bei erhöhten Lerndaten
  - ▶ Neuronales Netz in Matlab erstellen für bessere Übersichtlichkeit und Leistungsfähigkeit
- 

DEMO

# DATENBANK ERSTELLEN MIT FACESHIFT UND ANALYSEPROGRAM

VIELEN DANK FÜR IHRE  
AUFMERKSAMKEIT

