



# CoinCat

## Zwischenpräsentation

Tarik Glasmacher

Tim Bering

Bildverarbeitung WS22/23

Freiwilliges Projekt

# Agenda

- Überblick
  - Allgemein
  - Vorbereitung der Eingangsbilder
  - Neutrales Netzwerk
- Herausforderungen
- To-Do



# Überblick

# Überblick: Allgemein

- Ampelstatus: **Gelb**
  - Vorwiegender Grund: Herausforderungen bezüglich des neuronalen Netzwerkes
- Name abgeändert: *DINGUS* => **CoinCat**
- Weitere genutzte Toolboxes
  - *Neural Networking Toolbox*
  - *Parallel Processing Toolbox*





# Überblick: Vorbereitung der Eingangsbilder

- Anfängliches Problem: Hough-Algorithmus funktioniert nicht richtig bei perspektivisch verzerrten Bildern
  - Potenzieller Lösungsansatz: Wechsel auf Python / opencv (verworfen)
  - Tatsächlicher Lösungsansatz: Parameter angepasst
- Münzen werden erkannt
  - Nicht-Münzen-Kreise (siehe Torbogen auf dem 10€-Schein) werden sollen durch das neurale Netzwerk aussortiert werden
- Münzen werden (mit Rand) ausgeschnitten
  - Werden in eigenem Ordner abgespeichert
  - Werden auf Eingabegröße des neuronalen Netzwerkes gebracht



# Überblick: Neurales Netzwerk I

- Anfangs: Merkmalerkennung über SURF u.Ä.
  - Sehr ungenau
    - Potenzielle Fehlerquellen:
      - Hintergrund zu detailliert
      - Auflösung zu niedrig
      - Abgleich mit vorher abgespeichertem Referenzbild schwierig, vor allem bei perspektivisch verzerrten und abgeschnittenen Münzen
        - Ausrichtung an einer bestimmten Kante (siehe Gruppe Lyx / Florian) nicht möglich (auch wegen Rückseite der Münze)



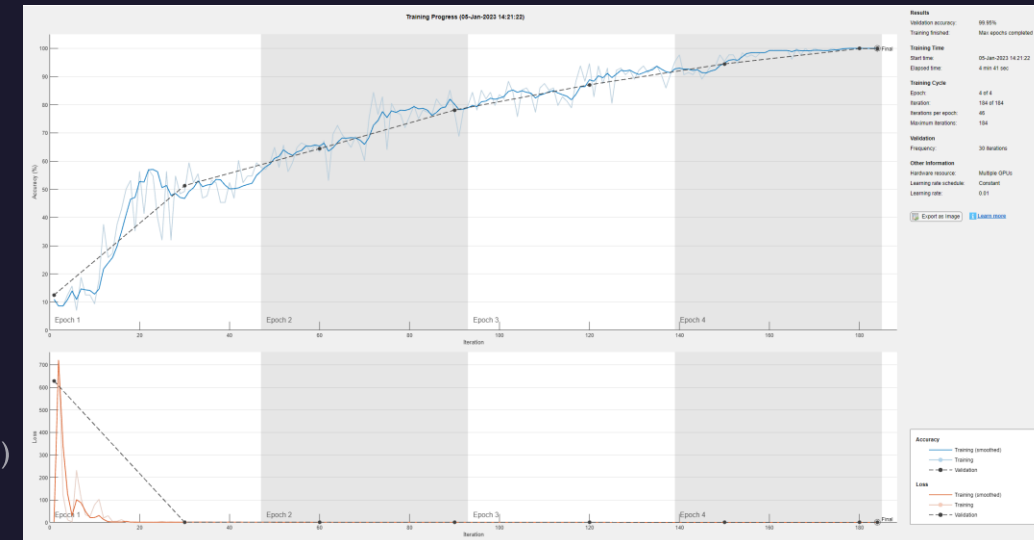
# Überblick: Neutrales Netzwerk II

- Verschiedene neurale Netzwerke ausprobiert
  - „Klassisches“ FF-Neural-Network
    - Zu generell
    - Nicht auf Bildverarbeitung zugeschnitten
  - Stacked Autoencoder
    - Training gibt keine / schlechte Ergebnisse
    - Ergebnisse werden beim Erstellen des Stacked-Netzes noch schlechter
  - **Image Classification / SGDM**
    - Ideal für den Sachzusammenhang
    - Dynamisches Hinzufügen von Trainings- und Validierungsdaten über *imageDatastore*



# Überblick: Neurales Netzwerk III

- Trainings- und Validierungsdaten werden aus einem Referenzbild gewonnen
  - Zufällige Rotation
- Anfangs: Genauigkeit von  $< 20.0\%$ 
  - Gründe: Weißer Hintergrund der Referenzbilder, Geringe Auflösung, Falsch eingestellte Parameter
- Deutliche Zeitkostenverringerung durch Parallel Processing
  - *Parallel Processing Toolbox* ermöglicht das Nutzen von bis zu 8 GPUs (Standardeinstellung) für das Training anstatt eines einzelnen CPU-Kernes
    - Parallel Pooling / Parpooling
  - Zeitersparnis: ca. 55.0% pro Training
- Aktuell: Genauigkeit von 99,95%
  - Epochen: 4
  - Validierungshäufigkeit: 30
  - 1.000 Bilder: 750 Trainingsbilder / 250 Validierungsbilder (Zuweisung zufällig aus den 1.000 Gesamtbildern)





# Herausforderungen

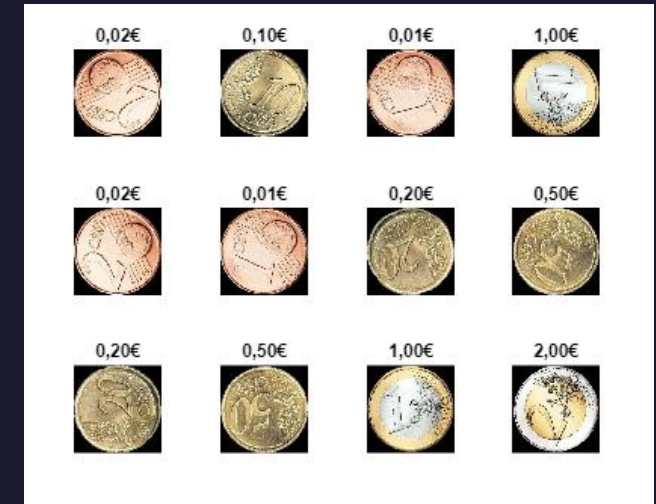
# Herausforderungen: Neurales Netzwerk

## 1. Einseitige Trainingsdaten

1. Trainingsdaten werden aus einem einzigen Referenzbild generiert
2. Aktuell: Ausschließlich zufällige Rotation des Referenzbildes
3. **Entstehendes Problem: Klassifiziert (fast) alle Daten aus dem Eingabebild als 0.10€**

## 2. Performance auf Mobilgeräten

1. Trainiertes Modell ist sehr groß
2. Übertragen / Laden des Modells im Transpiler
  1. Befehl `load serializedNet`
  2. Mitliefern der .m-Datei?
3. Potenziell sehr langsam / ressourcenintensiv



# To-Do

# To-Do

1. Trainings- und Validierungsdaten verbessern
2. Klassifikation anpassen
  1. Konfusionsmatrix anzeigen
  2. Konfusions-Schwellwert festlegen, ab welchem die Eingangsdaten als „Nicht valide“ markiert / ignoriert werden
3. Interne Programmstruktur verbessern
4. Entwicklung eines massentauglichen Logos
5. Applikation inklusive User-Interface
6. Testen





# Zusammenfassung

**CoinCat** kommt in der Entwicklung mit großen Schritten voran, jedoch stellten sich einige Herausforderungen in den Weg. Sobald diese überwunden sind, ist die Fertigstellung des Projektes in Sichtweite.



# Vielen Dank für Ihre Aufmerksamkeit!

Für Fragen und Anmerkungen stehen wir Ihnen nun zur Verfügung.

Tarik Glasmacher, Tim Bering

