# 2. Vorverarbeitung

* Abtastung und PCM
  + Digitalisierung = Diskretisierung + Quantisierung
    - Diskretisierung = messen an diskreten Punkten
    - Quantisierung = Umsetzung eines analogen Signals in ein digitales
      * Mehr Quantisierungsstufen => weniger Quantisierungsrauschen
  + Abtasttheorem: Abtastfrequenz muss mindestens doppelt so groß sein wie der höchste im Signal enthaltene Frequenzanteil
  + Pulse Code Modulation (PCM): Abtastung mit konstanter Abtastrate, Quantisierung der Abtastwerte und anschließende Codierung (i.d.R. im Binärcode)
* Digitale Audiodaten
  + Wesentlich bei periodischen Signalen: Amplitude und Periodenfrequenz f (bzw. Periodendauer 1/f bzw. Wellenlänge λ) -> f = c/ λ
  + Hörbereich des Menschen 20Hz - 20 kHz
  + Verlustfreie Audio-Codierung
    - Pulse Code Modulation (PCM)
    - DPCM (Differential PCM): Speicherung der Differenz zweier aufeinanderfolgender Abtastwerte
    - ADPCM (Adaptive DPCM): Vorhersage des zukünftigen Signalverlaufs und Speicherung der Differenz zur Vorhersage
* Digitale Bilddaten
  + Additive und subtraktive Farbmischung
    - RGB vs CMY
  + Weitere Farbmodelle: HSL und HSV
  + Bit pro Pixel wird Farbtiefe & Alphakanal (Transparenz)
* Vektorquantisierung
  + Transformation von kontinuierlichen Eingabevektoren auf ein endliches Klassenalphabet (Menge von Kodebuchklassen)
  + k-Means-Algorithmus
    - (zufällige) Auswahl von k Clusterzentren, z.B. durch zufällige Auswahl von k Eingabevektoren als Clusterzentren
    - Jedem Eingabevektor wird dem ihm am nächsten liegenden Clusterzentrum zugeordnet (Abstandsmaß: z.B. Euklidischer Abstand)
    - Es werden für jeden Cluster die Clusterzentren durch Schwerpunktbildung neu berechnet (Arithm Mittel errechnen aus Cluster Zentren & Vektoren)
    - Wiederholung: Falls sich nun die Zuordnung der Objekte ändert, weiter mit Schritt 2, sonst Abbruch
  + Anwendungen
    - Suche nach einheitliche Regionen in Bildern
    - Bestimmung von Laut-Unterklassen für die Spracherkennung
    - k-Means-Algorithmus ist Vorstufe des EM-Algorithmus
* Schwellwertoperationen & Histogramme
  + Punkt-Operationen: Die Werte von einzelnen Pixeln werden verändert, ohne dabei die Nachbarpixel zu betrachten
  + zB: Schwellwertoperationen, z.B. zur Transformation eines Grauwertbildes in ein Schwarzweißbild (Binärisierung)
  + Histogramm liefert zu jedem Grauwert (bzw. zu jeder Quantisierungsstufe der einzelnen Farbkanäle) dessen relative Häufigkeit
* Lineare Filter
  + A number and number in a square

    Description automatically generatedFilter berücksichtigen - anders als Punkt-Operationen - auch die Werte von Bildpunkten in der Umgebung des betrachteten Bildpunktes, um für einen Pixel im Originalbild einen neuen Wert zu berechnen
  + Mittelwertfilter (einfacher Weichzeichner)
  + A number and numbers on a white background

    Description automatically generatedFaltungsoperator ´\*´: paarweise Multiplikation der Wertepaare, die an derselben Position stehen, und Aufsummieren der Ergebnisse
  + A close-up of a number

    Description automatically generatedFilter als Faltungsoperation -> lineare Filter
  + Gauß-Filter (Gaußscher Weichzeichner)
  + Laplace-Filter zur Kantenhervorhebung
  + A white background with black text

    Description automatically generatedBoost-Filter (Scharfzeichner):
  + A screenshot of a white paper with yellow text

    Description automatically generatedKantenhervorhebung mit dem Sobel-Operator
  + Filter
    - Kein VZW -> Tief z.B. Gauß-Filter (Weichzeichner)
    - VZW -> Hoch z.B. Laplace-, Sobel-Filter (Kantenhervorhebung)
* Nichtlineare Operationen
  + Rangordnungsfilter
    - Sortiere die Pixel einer definierten Umgebung (z.B. 3x3, 11x11) aufsteigend nach ihrem Grauwert
    - ersetze den Grauwert des aktuellen Pixels (in der Mitte der Umgebung) durch den Wert an einer definierten Position in der Liste der Grauwerte:
      * Minimumfilter: 1. Position der Liste (dunkle Strukturen werden größer)
      * Medianfilter: mittlere Position der Liste Rauschen verschwindet, Ausreißer verschwinden
      * Maximumfilter: letzte Position der Liste (helle Strukturen werden größer)
  + Morphologische Operationen
    - Erosion: sobald der Mittelpunkt des Strukturelements auf Vordergrund trifft wird ein Vordergrundpunkt gezeichnet in der Mitte des Strukturelements --> Ränder werden nicht gezeichnet (abfräßen)
    - Dilatation (engl. Dilation): sobald der Mittelpunkt des Strukturelements auf Vordergrund trifft wird das komplette Strukturelement mit Vordergrundfarbe gezeichnet --> Ränder werden ausgedehnt
    - Opening: Erst Erosion, dann Dilatation
    - Closing: Erst Dilatation, dann Erosion
* Normierungsmaßnahmen
  + A diagram of a training process

    Description automatically generatedDie Normierung von Mustern soll den Wertebereich von Parametern, die für die Klassifikation irrelevant sind, reduzieren, um bei gegebenem Aufwand für die Klassifikation eine geringere Fehlerwahrscheinlichkeit zu erreichen

# 3. Merkmale

* Aufbau eines Klassifikationssystems
* Orthogonale Reihenentwicklung (DCT/DFT)
  + Die (zweidimensionale) DFT erlaubt die effiziente Implementierung von linearen Filtern, z.B. zur Bildvorverarbeitung
  + DFT dient zur digitalen Berechnung von Spektren und Spektrogrammen
  + A white paper with black and white squares and numbers

    Description automatically generatedDie DCT ist wesentlicher Bestandteil sowohl der JPEG-Bildkompression als auch der MP3-Audiokompression
  + orthogonal (rechtwinklig): Skalarprodukt verschiedener Basisvektoren jeweils 0
  + orthonormal, weil hierzu zusätzlich noch die Länge der Basisvektoren 1 sein müsste
  + DCT 🡪 siehe Bild (Achtung Faltung)
    - Koeffizienten der DCT sind dagegen immer reelle Zahlen
  + DFT: DCT kann als ein Spezialfall der DFT
    - DFT Koeffizienten: Real & Imaginärteil
  + DFT als auch die DCT setzen implizit voraus, dass das Eingangssignal periodisch
  + Daher bei der DCT keine impliziten Sprünge an den Rändern wie bei der DFT (siehe „Leck-Effekt“
    - DCT nimmt an, dass der Signalverlauf gespiegelt ist -> kein Sprung
  + DFT auf kurzen Abschnitten nichtperiodischer Signale, wie bei der Merkmalberechnung für die Spracherkennung, ist eine geeignete Fensterfunktion erforderlich (z.B. Hamming-Fenster). Man spricht hier auch von einer STFT (Short Time Fourier Transform)
  + STFT gilt die folgende Unschärferelation: Je höher die zeitliche Auflösung, desto geringer die Frequenzauflösung, und umgekehrt.
* Wavelet-Transformation
  + Kurzzeit-Fouriertransformation (STFT) erfordert Kompromiss zwischen Zeitund Frequenzauflösung
  + Häufig wünschenswert: bessere zeitliche Auflösung für hohe Frequenzen -> Wavelet-Transformation
  + einfachste Wavelet ist das Haar-Wavelet
  + Morlet-Wavelets: Es entsteht durch Modulation der Kosinusfunktion mit einer Gauß-Glocke
* Heuristische Verfahren
  + Heuristik bezeichnet in der Informatik eine Vorgehensweise, bei der man versucht, ein Problem zu Lösen mithilfe von Schätzungen, Faustregeln, intuitiv-intelligentem Raten
  + Nulldurchgangsrate
    - Die Nulldurchgangsrate berechnet sich als die Anzahl der Vorzeichenwechsel des Signals in einem Zeitfenster konstanter Länge, das über das Signal geschoben wird
* Merkmale für die Spracherkennung
  + Hörbereich: ca. 20 Hz – 20 kHz, höchste Schallempfindlichkeit bei 3–4 kHz
  + menschliche Ohr ist in der Lage, Schallwellen in ihre Frequenzanteile zu zerlegen
    - 25 Frequenzgruppen (kritische Bänder)
  + Diskrete Fourier Analyse (DFT) berechnet Frequenzspektrum von Abschnitt des Signals
  + Ziel: Beschreibung der spektralen Zusammensetzung des Sprachsignals
  + Verminderter Leck-Effekt durch Hamming-Fenster
  + Unschärfeprinzip: Je besser die Zeitauflösung, desto schlechter die Frequenzauflösung und umgekehrt
  + Spektrogramme
    - Zeit wird auf der x-Achse, die Frequenz in Hz auf der y-Achse
    - Intensität wird am Grad der Schwärzung
    - Breitbandspektrogramm
      * geringe Frequenzauflösung & hohe Zeitauflösung
    - Schmalbandspektrogramm
      * hohe Frequenzauflösung & geringe Zeitauflösung
    - Was sieht man im Spektrogramm?
      * Bereiche im Spektrum mit starker Intensität
      * besonders im Bereich von Vokalen
      * Harmonische der Anregungsfrequenz (Grundfrequenz)
    - Bandspektren
      * Frequenzgruppen, Bark- bzw. Mel-Skala wird durch Bank von Bandpassfiltern modelliert
      * Menschliches Gehör nimmt Frequenzen in Abschnitten wahr -> Bandspektren 25 Mel-Spektrumskoeffizienten die die Wahrnehmung abdeckt
* Merkmale für die Objekterkennung
  + häufig Merkmale von Teilbereichen des Bildes interessant, z.B. wenn das gesuchte Objekt nur einen Teil des Bildes überdeckt und dessen Position vorab nicht bekannt ist
  + auf Bildern eine gefensterte Fouriertransformation eingesetzt werden, um die Eigenschaften in lokalen Bereichen des Bildes zu ermitteln
  + Alternativ können Wavelets oder lineare Filteroperationen zur Merkmalberechnung verwendet werden (z.B. Gauß-Filter oder der Laplace-Operator)
  + Wichtige Rolle: Merkmale aus Farbe, Form (Kontur) und Textur (Oberflächenstruktur) sowie Punktmerkmale, häufig auch in Kombination
  + Punktmerkmale
    - Grundidee: Markante Punkte sind entscheidend die Objekterkennung
    - Die Eigenschaften dieser Punkte lassen sich durch Merkmalvektoren beschreiben
    - Merkmalvektoren sollten nach Möglichkeit invariant sein gegenüber: •Translation • Rotation • Skalierung • Änderung der Beleuchtung • Affiner Verzerrung (nur teilweise möglich)
    - Objekterkennung durch Zuordnung der markanten Punkte im gesuchten Objekt zu den markanten Punkten im Bild anhand ihrer Ähnlichkeit (z.B. Euklidischer Abstand) => Lage des Objekts im Bild
    - Bekannte Verfahren u.a.: SIFT (Scale-Invariant Feature Transform), SURF (Speeded Up Robust Features), ORB (Oriented FAST and Rotated BRIEF)
    - Alternative Anwendung: „Stitching“ mehrerer Bilder zu einem Panoramabild.
    - Keypoint-Detektoren, die markante Punkte identifizieren
    - Keypoint-Deskriptoren, die markante Punkte als Merkmalsvektor beschreiben -> Dieser Merkmal gehört zu diesem Punkt
    - Visual SLAM auf Basis von Punktmerkmalen SLAM: Simultaneous Localization And Mapping (Simultane Lokalisierung und Kartenerstellung

# 4. Numerische Klassifikation

* Die Komponenten des Vektors c sind reelle Zahlen, deshalb spricht man von numerischer Klassifikation
* Nichtparametrische Klassifikatoren
  + Speicherung der gesamten Stichprobe (z.B. Nächster-Nachbar-Klassifikator) -> Abstand zur Beobachtung
  + Zuweisung der Klasse des dem Merkmalvektor im Merkmalsraum am nächsten gelegenen Stichprobenelements
  + Nachteil: Rechenaufwand steigt mit Größe der Trainingsstichprobe
* Verteilungsfreie Klassifikatoren
  + Bestimmung der Parameter von geeigneten Trennfunktionen aus der Lernstichprobe
  + Es werden keine Annahmen über die statistische Verteilung der Merkmale getroffen
  + Support Vector Machine
    - Bestimme die breitestmögliche gerade „Straße“ zwischen den Klassengebieten. Der „Mittelstreifen“ bildet die Klassengrenze.
    - A graph of a line graph

      Description automatically generatedNur diejenigen Strichprobenelemente, die auf dem „Straßenrand“ liegen, beeinflussen den Verlauf der Klassengrenze. Sie werden Support Vectors (Stützvektoren) genannt
    - Durch Transformation des Merkmalraumes in einen Raum mit höherer Dimension werden auch verschachtelte Klassen linear trennbar
* Statistische Klassifikatoren
  + Kenntnisse über die statistischen Eigenschaften der Merkmalvektoren zu einer Klasse Ωκ sind gegeben
  + Die Dichtefunktion p(c | Ωκ) wird für die Klassifikation benötigt; diese wird dadurch bestimmt, dass deren unbekannten Parameter aκ aus einer geeigneten Stickprobe geschätzt werden.
  + Nur für einen Wertebereich (a < X < b) lässt sich eine Wahrscheinlichkeit angeben. Sie entspricht dem Integral der Dichtefunktion in den Grenzen von a bis b
  + Innerhalb des Problemkreises Ω wird zufällig eine Klasse ausgewählt, wobei die Klasse Ωκ mit der a priori Wahrscheinlichkeit p(Ωκ) gewählt wird.
  + Nach Wahl von Ωκ wird eine Beobachtung der Zufallsvariablen c gemacht (der Merkmalvektor eines Musters), wobei c die bedingte Dichte p(c | Ωκ) hat
  + Bayes-Formel
    - A math equation with black and red text

      Description automatically generatedGesucht ist diejenige Klasse Ωκ mit maximaler a posteriori Wahrscheinlichkeit p(Ωκ | c) -> Wahrscheinlichkeit, dass eine bestimmte Klasse vorliegt, wenn Merkmalsvektor c vorhanden ist
    - Dagegen lässt sich die bedingte Dichtefunktion p(c | Ωκ) in viele Fällen gut aus Stichproben abschätzen, ebenso wie die a priori Wahrscheinlichkeit p(Ωκ)
  + Bayes-Klassifikator
    - Nenner der Bayes-Formel unabhängig vom gesuchten Klassenindex κ und spielt deshalb bei der Bestimmung des Maximalwerts keine Rolle
    - Entscheidungsregel: Gegeben ein Merkmalvektor c, dann entscheide dich für diejenige Klasse Ωκ, für die gilt: Zähler = max
    - Bayes-Klassifikator der Klassifikator mit der geringsten Fehlerwahrscheinlichkeit
* Neuronale Netze
  + Perzeptron
    - Gewichtung und Bias für Merkmalsvektoren -> f(Sum(Gewicht\_i \* c\_i) – bias)
    - Als „Aktivierungsfunktion“ f dient eine sog. Schwellenwertfunktion: f (x) = 1 falls x ≥ 0, 0 sons
  + Training eines Perzeptrons
    - Ziel: Bestimme die Gewichte des Perzeptrons so, dass für eine gegebene Lernstichprobe die gewünschte Ausgabe erzeugt wird
    - Geeignete Gewichte lassen sich iterativ bestimmen
      * Beginnend mit einer zufälligen (oder uniformen) Initialisierung der Gewichte wird wiederholt
      * die Ausgabe y zu jedem Element der Lernstichprobe bestimmt
      * anhand der Differenz zu dem erwünschten Ergebnis eine Korrektur der Gewichte vorgenommen
    - Günstige Fälle Fehler am Ende des Trainings für alle Elemente der Lernstichprobe 0
      * Das gelingt aber nur bei linear separierbaren Merkmalsgebieten, da das Perzeptron lediglich eine lineare Trennebene modelliert
  + Mehrschichtperzeptron (Multilayer Perceptron, MLP)
    - Künstliche Neuronale Netze entstehen durch Verknüpfung von mehreren Perzeptrons zu einem Netzwerk (artificial neural network, ANN)
    - Häufigste Topologie: 3 (oder mehr) Schichten von Perzeptrons (Eingabeschicht, verborgene Schicht, Ausgabeschicht)
    - Training erfolgt i.d.R. über den sog. Backpropagation-Algorithmus
    - Selbst MLPs mit nur einer verborgenen Schicht können beliebig komplexe Funktionen und nichtlineare Klassengrenzen modellieren (Universal Approximation Theorem)
    - Künstliche neuronale Netze mit rekurrenten (= rückgekoppelten Kanten) sind sogar Turing-vollständig
  + Aktivierungsfunktionen
    - Nichtlineare Aktivierungsfunktionen sind zwingend notwendig, weil sich jedes Netz sonst durch Ausmultiplizieren der linearen Transformationen durch eine einzige Schicht darstellen ließe und damit nur noch lineare Funktionen modellieren könnte
    - Schwellwertfunktion: Biologisch plausibel (Neuronen feuern oder sie feuern nicht), aber mathematisch unhandlich weil kein Gradientenabstieg möglich
    - Die Sigmoidfunktion „quetscht“ alle reellen Zahlen in das Intervall ]0;1[
      * In tiefen Netzen führt sie zum Vanishing Gradient Problem
      * je mehr Schichten, desto weniger bleibt vom Gradienten übrig
    - Die ReLU-Funktion wird in tiefen neuronalen Netzen häufig verwendet und trägt dazu bei, das Vanishing Gradient Problem zu vermeiden
    - In der letztem Schicht eines NNs für die Klassifikation wird i.d.R. die Softmax-Aktivierungsfunktion verwendet. Sie liefert Schätzwerte für die a posteriori-Wkten aller Klassen
* Unüberwachtes Lernen
  + Beispiel: Transformation von kontinuierlichen Eingabevektoren auf ein endliches Klassenalphabet (Menge von Kodebuchklassen)
  + Jedem Vektor c wird der Index kt seiner Kodebuchklasse (oder Quantisiererzelle) zugeordnet (Vektorquantisierung)
  + Kodebücher können unüberwacht aus einer Stichprobe gelernt werden
  + Verfahren hierfür z.B.: • k-Means-Algorithmus • EM-Algorithmus
  + Gaußsche Mischverteilung (GMM)
    - mehrere Gauß-Dichten werden gewichtet und aufsummiert
    - Maximum-Likelihood-Schätzung der Parameter mit dem EM-Algorithmus
  + EM-Algorithmus (Expectation-Maximization)
    - Mit EM berechnete μ, σ
    - Problem bei der Schätzung der Parameter eines GMM: die Zuordnung eines Merkmalvektors x zu einer der M Dichten ist nicht bekannt
    - Der EM-Algorithmus löst das Problem durch iterative Optimierung (analog zum k-Means-Algorithmus).
    - Aber: es wird immer ein lokales Optimum gefunden, d.h. Parameter, die mindestens so gut sind wie die Startwerte
    - Erfahrung: es werden sehr gute Parameter gefunden, wenn die Startwerte gut sind und genug Trainingsdaten vorliegen
    - Steps:
      * Bestimme Startparameter B
      * Für Klassen i = 1, 2, 3, ….
        + Expectation-Schritt: Bestimme für jede Klasse Ωm und jeden Merkmalvektor xj die a-posteriori Wahrscheinlichkeit
        + Maximization-Schritt: Berechne die neuen Parameter B\_i

Gewichtungsfaktor a-priori-Wahrscheinlichkeit der Klasse m fließt mit rein

* + - * + Prüfe eine Abbruchbedingung

# 5. Spracherkennung

* Dynamic Time Warping (DTW)
  + Aufgabe: Einzelworterkennung
  + Idee: Nächster-Nachbar-Klassifikator für Wörter
  + Anlernphase ("Trainingsphase")
    - Aufnahme eines Referenzmusters zu jedem Wort im Wortschatz
    - Berechne zu jedem Referenzmuster die Folge von Merkmalvektoren und speichere diese
  + Laufender Betrieb
    - Aufnahme eines unbekannten Sprachmusters
    - Berechne zu diesem Sprachmuster die Folge von Merkmalvektoren
    - Bestimme nun den Abstand dieser Folge von allen Referenz-Folgen
    - A screenshot of a computer

      Description automatically generatedEntscheidung für das Wort mit dem geringsten Abstand
  + Problem: Folgen können unterschiedlich lang sein, Dehnungen und Stauchungen müssen nicht linear erfolgen, z.B. können einzelne Vokale stark gedehnt sein
  + DTW: Grundidee
    - Zuordnung von jedem Signalpunkt A zu jedem Signalpunkt B
  + Rückverzeigerung und Backtracking
    - Zeitliche Zuordnung kann ggf. durch Rückverzeigerung und Backtracking ermittelt werden
    - Bei jeder Minimierungsoperation speichert man in einem zusätzlichen n x m-Array einen Zeiger (bzw. einen geeigneten Hinweis) auf das Vorgängerfeld, das den minimalen Kostenbeitrag geliefert hat (Rückverzeigerung)
    - Nach Abschluss der Iteration lässt sich der optimale Pfad ausgehend vom Feld [n,m] rückwärts rekonstruieren (Backtracking)
  + DTW ist ein klassischer Algorithmus zur Einzelworterkennung
  + Besonders geeignet für sprecherabhängige Erkennung von Wörtern, zu denen nur ein einziges akustisches Referenzmuster vorliegt
* Hidden-Markov-Modelle (HMM)
  + gegeben: Menge von Wörtern W = {W1,...,WL}
  + beobachtet wird eine Merkmalvektorfolge X = x1,...,xT (Äußerung)
  + A yellow and black text

    Description automatically generatedwelches Wort wurde gesprochen?
  + a-priori-Wkt. der Wörter P(W\_l ) werden z.B. durch Auszählen einer Stichprobe geschätzt
  + aber: a-posteriori Wkt. P(X | W\_l ) kann durch Hidden-Markov-Modelle (HMM) geschätzt werden

🡪 Model mit Reihen von Zuständen, wo man weiß mit welcher Wskt. Ein Buchstabe im Wort ausgesprochen wird (Summe aller ausgehenden Zweige eines Zustandes = 1)

* + Bei mehreren möglichen Pfaden werden die Wahrscheinlichkeiten der einzelnen Pfade aufsummiert
  + HMM-Topologien
    - Anzahl der Zustände => typischerweise 3 Mal so viel wie Laute: haben (5 Laute) --> 15 Zustände
    - Keine Rückschritte, da Vertauschungsgefahr
  + Berechnung der Produktionswahrscheinlichkeit von einem Wort
    - Effizienter: berechne Vorwärts- und Rückwärtswahrscheinlichkeiten
    - Wie wahrscheinlich bin ich zum Zeitpunkt t im Zustand J und habe bis dahin die Merkmalsvektoren x\_1 bis x\_t produziert
    - Idee: Zusammenfassen und Aufsummieren der verschiedenen Varianten wie man in einen Zustand landet
  + Vorwärtsalgorithmus and Rückwärtsalgorithmus
    - Für Erkennung benötigt man Vorwärtsalgorithmus; Für Training zusätzlich auch noch Rückwärts
    - Viterbi-Algorithmus
      * ist eine Modifikation des Vorwärtsalgorithmus, der im Rekursionsschritt anstelle der Summe eine Maximierung durchführt
      * Zur Bestimmung der besten Zustandsfolge werden wird (wie beim Dynamic Time Warping) eine Rückverzeigerung und Backtracking verwende
      * A whiteboard with writing and a diagram

        Description automatically generatedWir suchen die wahrscheinlichste Zustandsfolge, die unsere Beobachtung X erzeugt hat

# 6. Objekterkennung

* Kantendetektion
  + Canny Alg (Erkennung alle tatsächlichen Kanten sollen gefunden werden mit Breite 1 px, aber keine falschen)
    - Farbbild
    - -> Grauwertbild
    - -> Sobel-Operator
    - -> Nicht Max Unterdrückung (winkel = arctan(G\_y/G\_x) Sobel)
    - -> Hysterese-Schwellwertverfahren (Schwellwerte T\_1 <= T\_2)
* Hough-Transformation (Hough-Raum (r,θ))
  + Jeder Geraden im Bildraum entspricht ein Punkt im Hough-Raum (r->y; winkel->x)
  + r = xi ⋅ cosθ + yi ⋅sinθ
  + A math equation with black text

    Description automatically generatedJedem Punkt im Bildraum entspricht eine sinusförmig verlaufende Kurve im Hough-Raum
  + Jedem Kurven-Schnittpunkt im Hough-Raum entspricht eine Gerade durch mehrere Punkte im Bildraum
  + Erkennen von Kreisen -> Hough-Raum (x,y,r) mit Kreismittelpunkten und Radien
  + Hoher Rechenaufwand und Speicherbedarf
* Histogrammbasierte Objekterkennung
  + Objekterkennung und -lokalisierung ausschließlich auf Basis des Farbhistogramms
  + Vergleich zweier Histogramme I (unbekanntes Bild) und M (Modell) erfolgt durch Berechnung des Histogrammschnitts (histogram intersection)
* Viola-Jones-Algorithmus
  + Detektion von Gesichtern
  + später auch erfolgreich zur Detektion anderer Objekte oder zur Erkennung von Personen (ganzer Körper)
  + Training (Stichprobe von Bildern einheitlicher Größe -> Trennung Positivbeispiele von Negativbeispielen)
  + Kaskade von Klassifikatoren
  + Effiziente Merkmalsberechnung
* CNN (Convolutional Neural Networks)
  + dominierende Verfahren zur Objekterkennung/Klassifikation von Bildern
  + Das ConvNet erlernt neben den Klassengrenzen auch eine geeignete Merkmalberechnung anhand der Trainingsstichprobe
  + Es kann daher mit rohen Bilddaten trainiert werden
  + Schichten
    - obersten Schichten eines ConvNet führen Faltungsoperationen (engl. Convolution) durch
    - Nichtlinearität (Schwellwertfunktion, oft ReLU)
    - Pooling- oder Sub-Sampling-Schicht -> große Feature map zu kleinere (zB: Zusammenfassen von 4 Pixel zu einem)
    - Fully Connected Layer (zur Klassifikation, MLP, vgl. Kapitel 4)
  + Gesichtserkennung
    - Merkmalvektor zu dem Bild eines Gesichts wird oft als Face Embedding
    - Eigenschaft: gleiche Person möglichst ähnliche Merkmalsvektoren -> verschiedene Personen möglichst unterschiedliche Merkmalsvektoren
    - Triple Loss min: je 3 Bilder betrachtet, davon 2 von der gleichen Person (Anchor und Positive) sowie eins von einer anderen Person (Negative)
    - Abstand zwischen Anchor und Positive größer als der zwischen Anchor und Negative -> Anchor & Positive möglichst klein
* Objektverfolgung
  + Merkmale für die zu verfolgende Region: häufig Farbhistogramme, aber auch Texturmerkmale etc.
  + Meist Kernel-Tracking mit rechteckiger Kernelfunktion (d.h. ein rechteckiger Bildausschnitt wird verfolgt)
  + Klassisches Verfahren zum Kernel-Tracking: Mean-Shift-Algorithmus
  + TLD-Tracking
    - Verfolgung über einen längeren Zeitraum erfordert Möglichkeit, verloren gegangenes Zielobjekt wiederzufinden („Detection“) und das Tracking neu zu initialisieren
    - während des Tracking lernt das Model, wie das Objekt aussieht angepasst an die Umgebungen
  + Prognose der zukünftigen Objektposition
    - Kalman-Filter: Eine Hypothese wo sich das Objekt befindet
    - Partikelfilter: Parallele Verfolgung mehrerer Hypothesen über das Objekt

# 7. Experimentelle Evaluation

* Stichprobe
  + Lernstichprobe: Grundlage für automatische Parameterschätzung
  + Validierungsstichprobe: Grundlage für das Optimieren von einstellbaren Parametern (z.B. Schwellwerte u. Gewichte)
  + Teststichprobe
    - darf erst dann betrachtet werden, wenn die Optimierung abgeschlossen ist
    - dient der Ermittlung von Erkennungsraten, die realistisch sind für ungesehenen Daten
* Gütemaße
  + Vierfeldertafel
    - FP: False positives, fälschlicherweise als „positiv“ erkannt, falsch positiv, Fehler II. Art
    - FN: False negatives, fälschlicherweise als „negativ“ erkannt, falsch negativ, Fehler I. Art
  + Recall und Precision
    - Recall = Sensitivität = Trefferquote
    - Precision = Genauigkeit = Positiver Vorhersagewert
  + Sensitivität und Spezifität
    - Sensitivität = Recall
    - Spezifität = true negative rate -> Vorsicht Marketing
  + Negativer und positiver Vorhersagewert
    - Negativer Vorhersagewert = Segreganz = Trennfähigkeit
    - Positiver Vorhersagwert = Precision
  + Falsch-Positiv- und Falsch-Negativ-Rate
    - Falsch-Positiv-Rate = false alarm rate
    - Falsch-Negativ-Rate = false reject rate
  + Accuracy und F-Wert
    - Accuracy = Erkennungsrate
    - F-Wert = F1 score = F-score = F-measure
  + F-Wert besitzt eine höhere Aussagekraft, ist aber beim Vergleich von Klassifikatoren mit unterschiedlichen Arbeitspunkten dennoch mit Vorsicht zu genießen
  + Ziel in praktischen Anwendungen ist i.d.R. nicht die Minimierung der Fehlerrate, sondern die Minimierung der „Kosten“
  + Area under ROC Curve: Fläche unter der Kurve: Wahrscheinlichkeit, dass die Klassifikation richtig ist
  + Fehlermaße für mehr als 2 Klassen
    - Erkennungsrate = recognition rate = rr
  + Jaccard-Index
    - Kennzahl, mit der sich ganz allgemein die Ähnlichkeit zweier Mengen bestimmen lässt
    - Berechnung über die Größe der Flächen (Anzahl der Pixel)
    - Berechnet sich als die Größe der Schnittmenge geteilt durch die Größe der Vereinigungsmenge