

Dimensionsreduktionsverfahren in der computergraphischen Fluidsimulation

Ludwig Schmutzler

Dresden, 16.09.2010

Gliederung

1. Motivation/Zielstellung
2. Grundlagen
- 3. Hauptkomponentenanalyse (PCA)**
- 4. Kern-PCA**
- 5. Simulation**
6. Ergebnisse
7. Diskussion
8. Demonstration
- Quellen
- Anhang

01 Motivation/Zielstellung

- nicht selten hochkomplexe Beschreibungen von Naturphänomenen in der Computergraphik
- typischerweise große Datenmengen (z.B. Zeitreihe einer abgetasteten Fluidoberfläche über 128^3 -Gitter: $\sim 7,8$ GiB bei 1000 Zeitschritten und 4 Byte floats)
- Finden von Strukturen in der Datenflut, um dieselben Phänomene mit weniger Variablen beschreiben zu können
- kann durch Dimensionsreduktionsverfahren erreicht werden

01 Motivation/Zielstellung

Ziel der Belegarbeit:

Anwendung von Dimensionsreduktionsverfahren auf **Fluidgrenzflächen** analog Treuille [TLP06]
zur **Datenkompression und Animation**

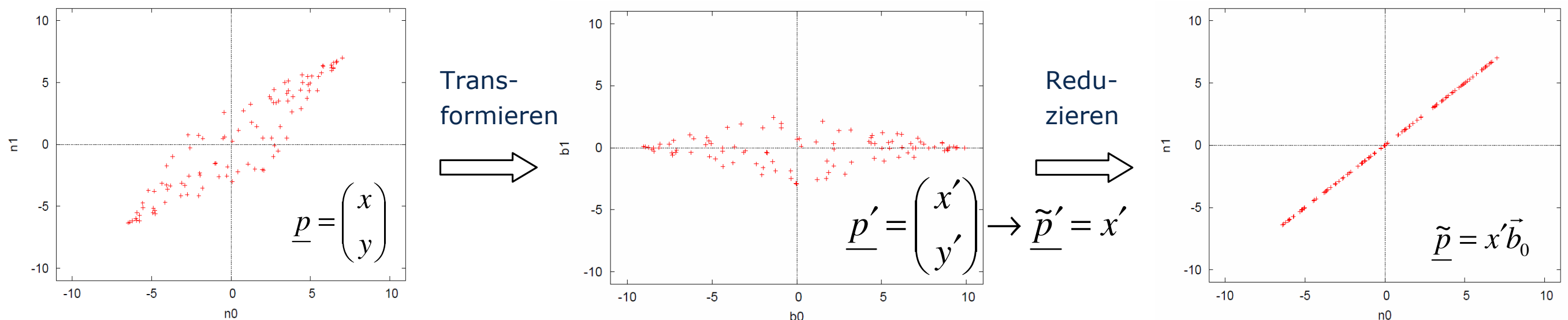
ausgewählt wurden:

- Hauptkomponentenanalyse (PCA) als lineares
- Kern-PCA als nichtlineares Verfahren

01 Motivation/Zielstellung

Grundidee der Reduktionsverfahren:

- transformiere Daten in ein Koordinatensystem, in welchem ihre Beschreibung mit weniger Komponenten auskommt
- PCA erreicht dies, indem Richtungen der größten Varianzen gefunden werden, sie heißen **Hauptkomponenten**
- Reduktion besteht im Vernachlässigen von niederwertigen Richtungen



02 Grundlagen

Oberflächen in der Fluidsimulation - Level sets

- Eingabedaten in Form von sog. **level sets** gegeben
- vergleichbar mit Definition für impliziten Kurven/Flächen[OF02]:

- gegeben eine implizite Funktion ϕ deren Nullstellenmenge $\delta\phi$ die Isokurve/fläche erklärt:

$$\delta\phi \equiv \{\vec{x} \mid \phi(\vec{x}) = 0\}$$

- Unterschied zum level set: Isokurve/fläche muss hier nicht zwangsläufig Nullstellenmenge sein:

$$\delta\phi_{\Delta} \equiv \{\vec{x} \mid \phi(\vec{x}) = \Delta\}$$

2	2	2,2	2,8	3,1
1	1	1,4	2	2
0	0	0,7	1	1,4
1	0,7	0	0	0,7
2	1	1	0,7	0

Datensätze als abgetastete Skalarwerte über einem uniformen Gitter (eingetragen der Fluidrand)

03 Hauptkomponentenanalyse (PCA)

- wohlbekanntes mathematisches Verfahren, Ursprünge im ausgehenden 19.Jh./frühen 20.Jh., v.a. durch Pearson und Hotelling [Jo102]
- Stärken spielten sich mit Verfügbarkeit von leistungsfähigen Rechnern aus, damit konnte Verfahren auf multidimensionale Daten angewendet werden
- lineares Verfahren, universell anwendbar, durch Orthogonalität einfache Hin- und Rückabbildung in bzw. aus reduzierten Raum möglich
- Finden der Hauptkomponenten über Eigenwertzerlegung, sodass Nebenabhängigkeiten in den Daten minimiert werden (lassen sich über die **Kovarianz** beschreiben)
- nach der Transformationen in Bildraum sind **Varianzen** maximiert und **Kovarianzen** minimiert

03 Hauptkomponentenanalyse (PCA)

Der PCA-Algorithmus [Shl03]

- Daten werden als Spaltenvektoren einer großen Matrix \mathbf{X} aufgefasst:

$$\mathbf{X} = (\vec{x}_0 \dots \vec{x}_t \dots \vec{x}_{T-1})$$

- Aufstellen der Kovarianzmatrix

$$\mathbf{C}_X \equiv \frac{1}{T-1} \mathbf{X} \mathbf{X}^T \quad (\text{bei mittelwertbereinigten Daten})$$

- Lösen des Eigenwertproblems:

$$\lambda \vec{b} = \mathbf{C}_X \vec{b}$$

- Mittel: SVD

$$\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \quad \mathbf{V} \text{ enthält Eigenvektoren } \vec{b} \text{ von } \mathbf{C}_X \text{ für } \mathbf{A} \equiv \left(\sqrt{T-1}\right)^{-1} \mathbf{X}^T$$

03 Hauptkomponentenanalyse (PCA)

Reduzieren der Basis – Verwerfen von Eigenvektoren:

- Die ersten M Eigenvektoren werden als **Hauptkomponenten** bezeichnet und bilden die reduzierte Basis **B**

Merkmalsextraktion:

- Projektion der Datenvektoren auf Hauptkomponenten ergibt Merkmalsmatrix A:

$$A = (\vec{a}_0 \dots \vec{a}_{T-1}) = B \cdot X \quad \dim \vec{a}_t < \dim \vec{x}_t$$

Merkmale beschreiben die Datenvektoren im transformierten (reduzierten) Raum

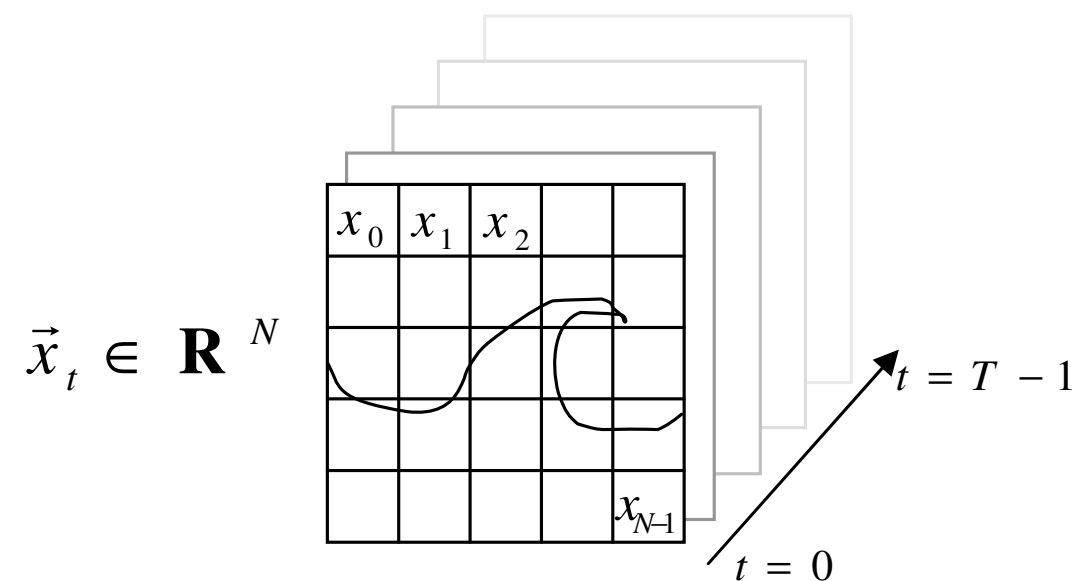
Rekonstruktion:

$$\tilde{X} = B^T \cdot A$$

$$B^T = B^{-1}$$

(aufgrund der Orthogonalität)

03 Kompression mithilfe der PCA



\vec{x}_t ... Datenvektor mit N Einträgen einer
Zeitreihe mit T Zeitschritten

Speicherverbrauch im unkomprimierten Fall:

$$S \propto N \cdot T$$

\uparrow
 Dimension d.
Datensatzes

\uparrow
 Anzahl der
Datensätze

03 Kompression mithilfe der PCA

Speicherverbrauch im komprimierten Fall:

1. reine Merkmalsextraktion:
(ohne Speicherung d. Hauptkomponenten)

$$S \propto M \cdot T$$

M... Dimension eines Merkmalsvektors ($M < N$)

2. mit Rekonstruktion:
(inkl. Speicherung d. Hauptkomponenten)

$$S \propto M \cdot (T + N)$$

Kompressionsgrad K:

$$K = N / M$$

$$K \approx T / M$$

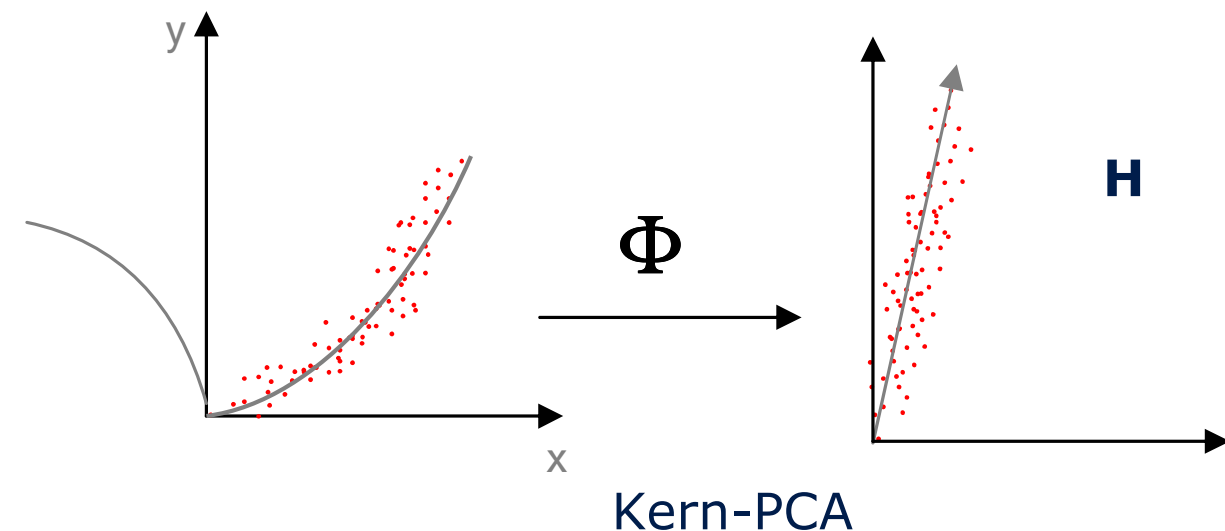
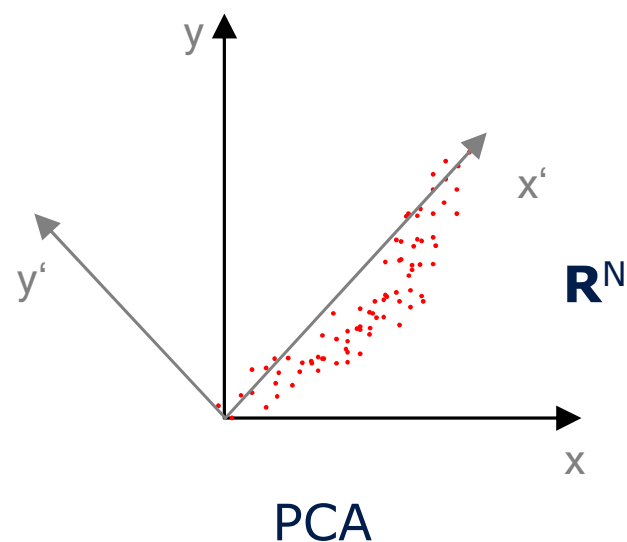
(wenn räumliche Auflösung wesentlich
größer als zeitliche, d.h. $N \gg T$)

04 Kern-PCA

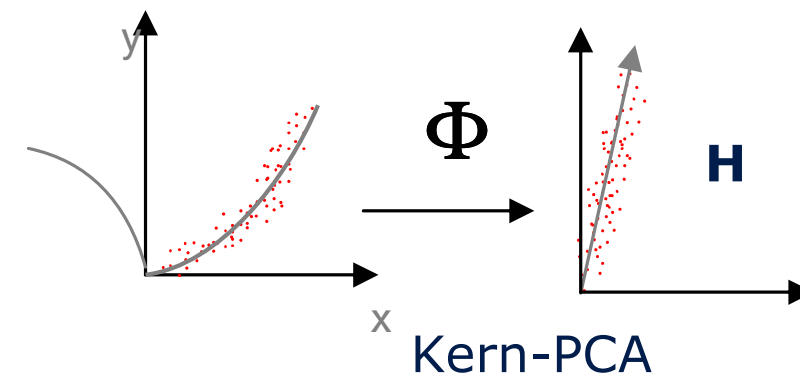
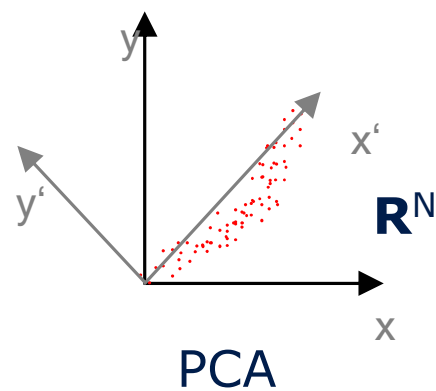
- verknüpft Kernmethoden mit Mitteln der PCA, um nichtlineare Muster zu finden
Grundlage bildet Abbildung Φ , welche x in hochdimensionalen Raum \mathbf{H} abbildet:

$$\begin{aligned}\Phi: R^N &\rightarrow H \\ \vec{x} &\rightarrow \Phi(\vec{x})\end{aligned}$$

- Φ ist dabei in seiner Komplexität unbeschränkt
- durch Entzerrung werden nichtlineare Strukturen im Eingaberaum in \mathbf{H} linear, sodass PCA-Algorithmus optimal ansetzen kann



04 Kern-PCA



$$X = (\vec{x}_0 \dots \vec{x}_t \dots \vec{x}_{T-1})$$

$$C_X \equiv \frac{1}{T-1} X X^T$$

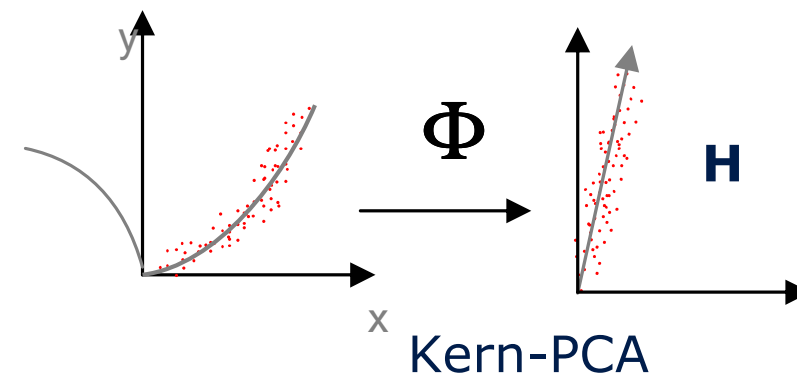
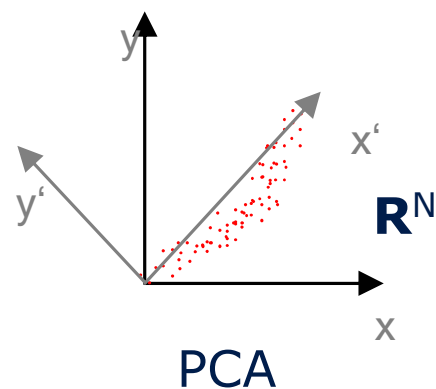
$$\lambda \vec{b} = \lambda C_X$$

$$\hat{X} = (\Phi(\vec{x}_0) \dots \Phi(\vec{x}_t) \dots \Phi(\vec{x}_{T-1}))$$

$$\hat{C}_X \equiv \frac{1}{T-1} \hat{X} \hat{X}^T$$

$$\lambda \vec{b} = \lambda \hat{C}_X$$

04 Kern-PCA



$$X = (\vec{x}_0 \dots \vec{x}_t \dots \vec{x}_{T-1})$$

$$C_X \equiv \frac{1}{T-1} X X^T$$

$$\lambda \vec{b} = \lambda C_X$$

$$\hat{X} = (\Phi(\vec{x}_0) \dots \Phi(\vec{x}_t) \dots \Phi(\vec{x}_{T-1}))$$

$$\hat{C}_X \equiv \frac{1}{T-1} \hat{X} \hat{X}^T$$

Nicht explizit möglich!



Kerne, Kernmatrix

$$\lambda \vec{b} = \lambda \hat{C}_X$$

04 Kern-PCA

Problem: potenzielle Unbeschränktheit von \mathbf{H} und Φ , explizites Ausrechnen nicht möglich

Trick: Kernfunktionen

- PCA lässt sich allein auf Grundlage von Skalarprodukten beschreiben
- Definition sog. Kernfunktionen, die Skalarprodukte zweier Vektoren aus \mathbf{H} in \mathbf{R}^N berechnen können:

$$k(\vec{x}, \vec{y}) = \langle \vec{\Phi}(\vec{x}), \vec{\Phi}(\vec{y}) \rangle$$

Klassen von Kernen [SMS99][YZZL08]:

- polynomielle Kerne: $k(\vec{x}, \vec{y}) = \langle \vec{x}, \vec{y} \rangle^d$
- Sigmoid-Kerne: $k(\vec{x}, \vec{y}) = \tanh(\kappa \langle \vec{x}, \vec{y} \rangle + \theta)$
- Kerne für Gauß'sche RBF: $k(\vec{x}, \vec{y}) = \exp(-\|\vec{x} - \vec{y}\|/(2\sigma^2))$

04 Kern-PCA

Die Kernmatrix **K** [SS01]

- **K** ist Gram'sche Matrix in **H** [SA94]
(die Matrix aller N^2 Einträge zu einer Bilinearform, z.B. Skalarprodukt, definiert auf N-dimensionalem Körper)
- Einträge sind die paarweisen Skalarprodukte der Bildvektoren in **H**, welche sich über die Kernfunktionen in **R^N** berechnen lassen

$$K = \begin{pmatrix} \langle \vec{\Phi}(\vec{x}_0), \vec{\Phi}(\vec{x}_0) \rangle & \cdots & \langle \vec{\Phi}(\vec{x}_0), \vec{\Phi}(\vec{x}_{T-1}) \rangle \\ \vdots & \ddots & \vdots \\ \langle \vec{\Phi}(\vec{x}_{T-1}), \vec{\Phi}(\vec{x}_0) \rangle & \cdots & \langle \vec{\Phi}(\vec{x}_{T-1}), \vec{\Phi}(\vec{x}_{T-1}) \rangle \end{pmatrix} = \begin{pmatrix} k(\vec{x}_0, \vec{x}_0) & \cdots & k(\vec{x}_0, \vec{x}_{T-1}) \\ \vdots & \ddots & \vdots \\ k(\vec{x}_{T-1}, \vec{x}_0) & \cdots & k(\vec{x}_{T-1}, \vec{x}_{T-1}) \end{pmatrix}$$

Skalarprodukt in **H** kann in **R^N** berechnet werden

04 Kern-PCA

Der (Um-)Weg

- Eigenwertzerlegung der Kovarianzmatrix **C** entspricht Eigenwertzerlegung von **K** [SS01]:

$$\lambda \vec{b} = \lambda \hat{C}_x \Rightarrow \tilde{\lambda} \vec{\alpha} = K \vec{\alpha}$$

(die erhaltenen Eigenvektoren α stimmen bis auf konstanten Koeffizienten mit Hauptkomponenten b überein, müssen noch umnormalisiert werden $\rightarrow \alpha^*$)

- Reduktion besteht wieder im Verwerfen von Eigenvektoren
- durch Mächtigkeit von Φ kann im Bildraum **H** höhere Abstraktion und damit größere Reduktion erreicht werden [SMS99] als bei PCA, Wahl des Kernes jedoch abhängig vom Inhalt der Daten

Merkmalsextraktion:

Merkmalsvektoren ergeben sich wieder analog PCA:

$$A = (\vec{a}_0 \dots \vec{a}_{T-1}) = \begin{pmatrix} \vec{\alpha}_0^* \\ \vdots \\ \vec{\alpha}_{M-1}^* \end{pmatrix} \cdot K$$

04 Kern-PCA

Probleme im Hochdimensionalen

- Mittelwertbereinigung nicht direkt möglich, aber indirekt über Modifikation der Kernmatrix [SS01]:

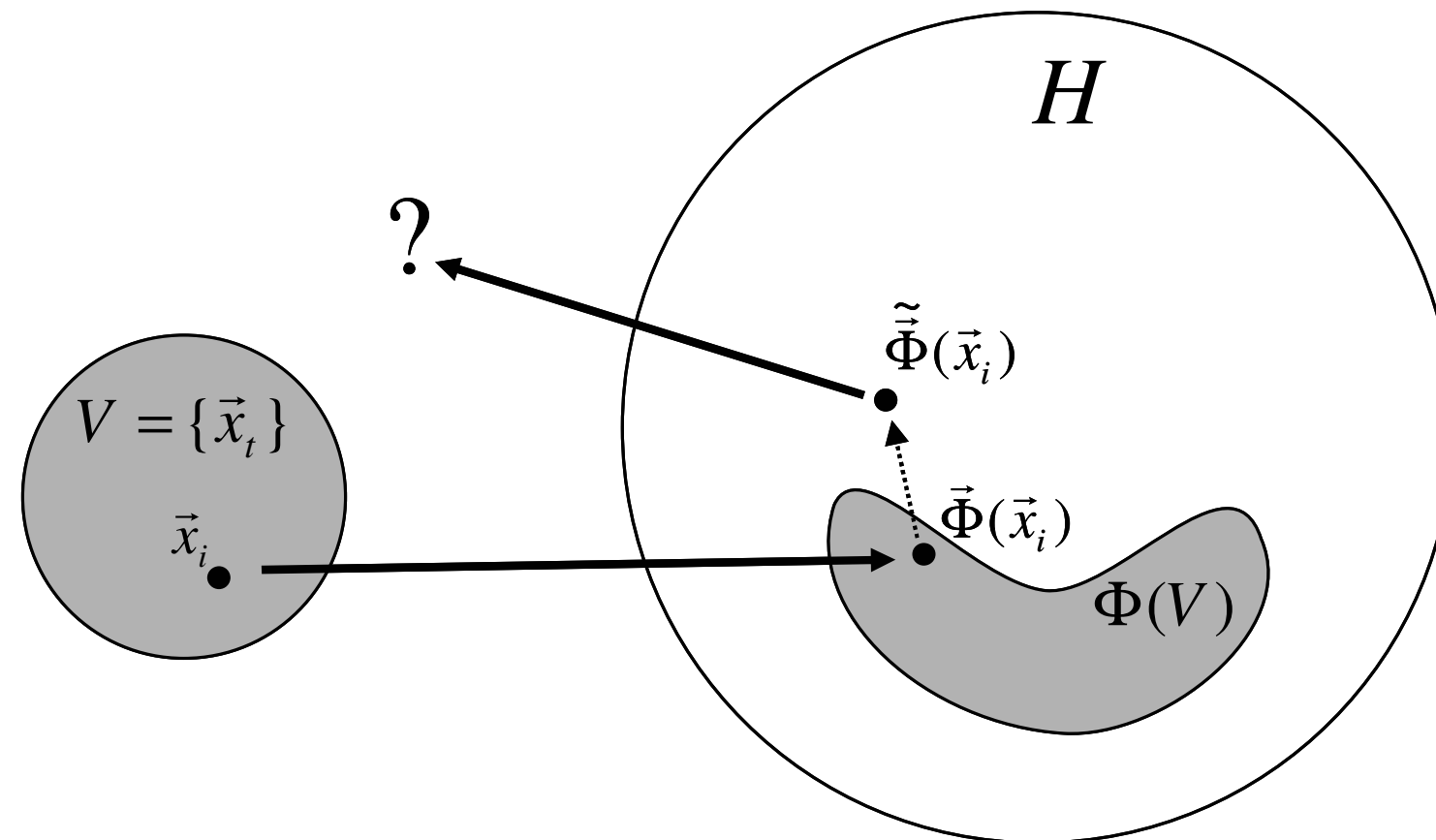
$$\tilde{K} = K - \frac{1}{T} EK - \frac{1}{T} KE + \frac{1}{T^2} EKE$$

E... Einheitsmatrix

- Eigenwertzerlegung mit modifizierter Kernmatrix entspricht dann der Eigenwertzerlegung der Kovarianzmatrix in **H** nach vorheriger Mittelwertbereinigung

04 Kern-PCA

Urbild-Problem [SS01]:



Rekonstruktion ist
Linearkombination:

$$\tilde{\Phi}(\vec{x}_i) = \sum_{i=0}^{M-1} a_i \vec{b}_i$$

04 Kern-PCA

(Exakte) Rekonstruktion

- analog PCA: Abbildung der Merkmale auf Hauptkomponenten.
- Anschließend Anwendung einer invertierbaren Funktion $f_k(\langle \vec{x}, \vec{y} \rangle) = k(\vec{x}, \vec{y})$

für $\tilde{\Phi}(\vec{x}_i) = \sum_{i=0}^{M-1} \hat{a}_i \vec{b}_i = \Phi(\vec{z})$ ergibt sich für z:

$$\vec{z} = \sum_{i=0}^{N-1} \langle \vec{z}, \vec{e}_i \rangle \vec{e}_i = \sum_{i=0}^{N-1} f_k^{-1}(k(\vec{z}, \vec{e}_i)) \vec{e}_i = \sum_{i=0}^{N-1} f_k^{-1} \left(\sum_{j=0}^{T-1} \hat{\alpha}_j k(\vec{x}_j, \vec{e}_i) \right) \vec{e}_i \quad \text{mit} \quad \hat{a}_j = \sum_{i=0}^{M-1} a_{t,i} \tilde{\alpha}_{i,j}$$

Für bestimmte Gruppen von Kernen möglich (polynomielle ungeraden Grades sowie Sigmoid-Kerne)

PROBLEM: Mittelwertfrei? Vor der Anwendung der Umkehrabbildung müssten die Mittelwerte rückgerechnet werden, diese sind jedoch niemals explizit ausgerechnet worden

→ Weg: Urbildapproximation, z.B. durch senkrechte Projektion von $\tilde{\Phi}(\vec{x}_i)$ auf $\text{span } \Phi(\vec{z})$ [SS01]

05 Simulation

Ursprünglicher Ansatz:

Modellreduktion analog Treuille über Galerkin-Projektion:

- Neben Datenvektoren wird Zeitentwicklungsfunktion des Modells in reduzierten Raum überführt:

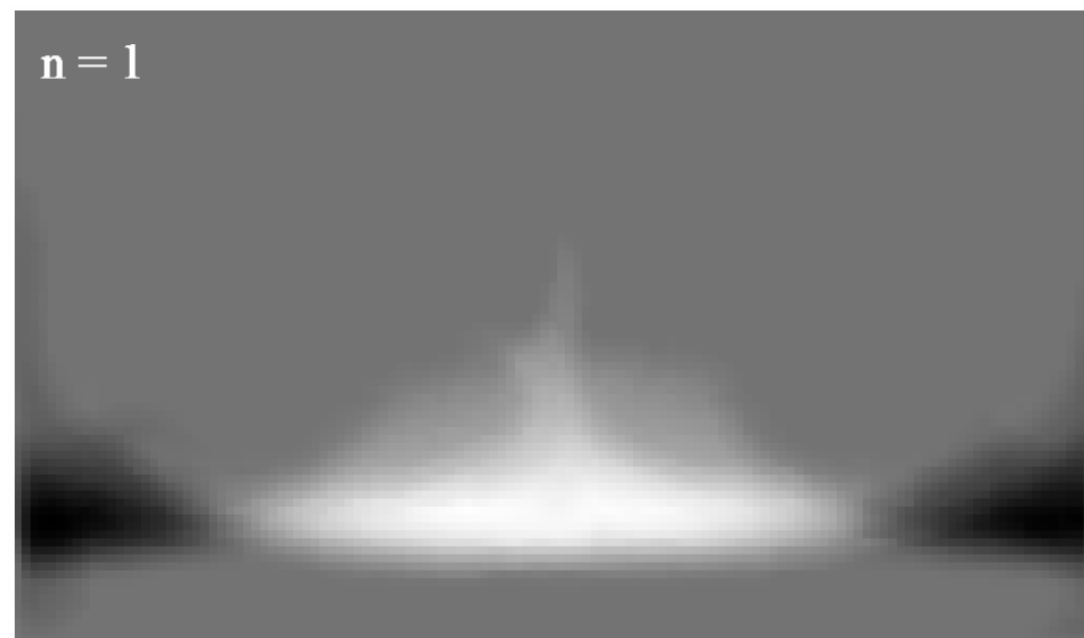
$$\dot{x} = F(\vec{x}) \quad \Rightarrow \quad \dot{\tilde{x}} = \tilde{F}(\tilde{x}) \quad \text{mit} \quad \begin{aligned} B : \vec{x} &\rightarrow \vec{a}, \quad B^{-1} : \vec{a} \rightarrow \vec{x} \\ \tilde{F} &= B \circ F \circ B^{-1} \end{aligned}$$

- Im Rahmen der Belegarbeit ist Zeitentwicklungsfunktion nicht gegeben, sondern reines *level set*

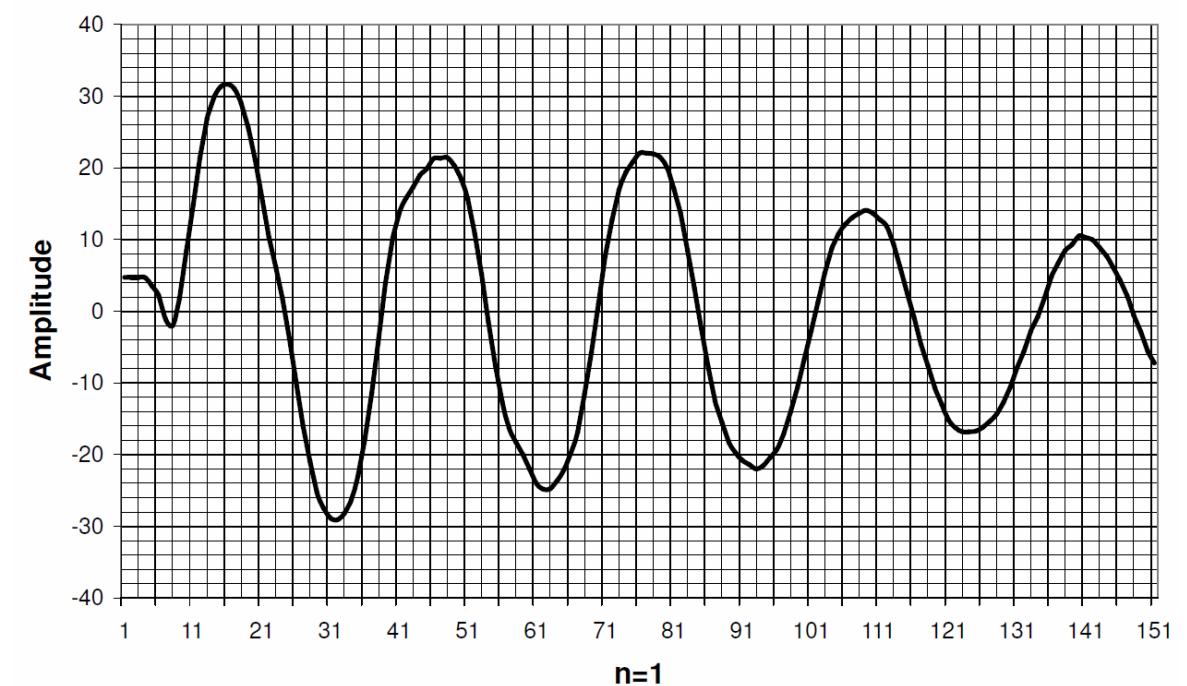
05 Simulation

Ein einfaches Modell:

- Grundlage ist die Interpretation der Zeitfunktionen der Merkmale bei bestimmten Datensätzen
- Hauptkomponenten beschreiben die Orte der Dynamik und Merkmale die Intensität
- Interpretation als Moden, Annäherung der Merkmalsverläufe mit gedämpften Sinusschwingungen [Kut04]



Visualisierung der 1. Hauptkomponente
eines Datensatzes



Verlauf des zugehörigen Merkmals

05 Simulation

→ Approximation der i-ten Merkmalsentwicklungsfunktion durch Summen gedämpfter (harmonischer) Partialschwingungen (d.h. ganzzahlige Vielfache der Grundschwingung)

Modellformel:

$$\forall i = 0..M-1: y_i(t) = o_i + e^{-t\delta_i} \cdot \hat{y}_i \cdot \sum_{n=0}^{n_i-1} e^{-n\beta_i} \sin(2\pi \cdot n f_i t + \varphi_i)$$

Anzahl der Oberschwingungen
Offset, Amplitudendämpfung, Maximalamplitude, spektrale Dämpfung, Grundfrequenz, Phasenlage

Approximation eines rekonstruierten Datenvektors:

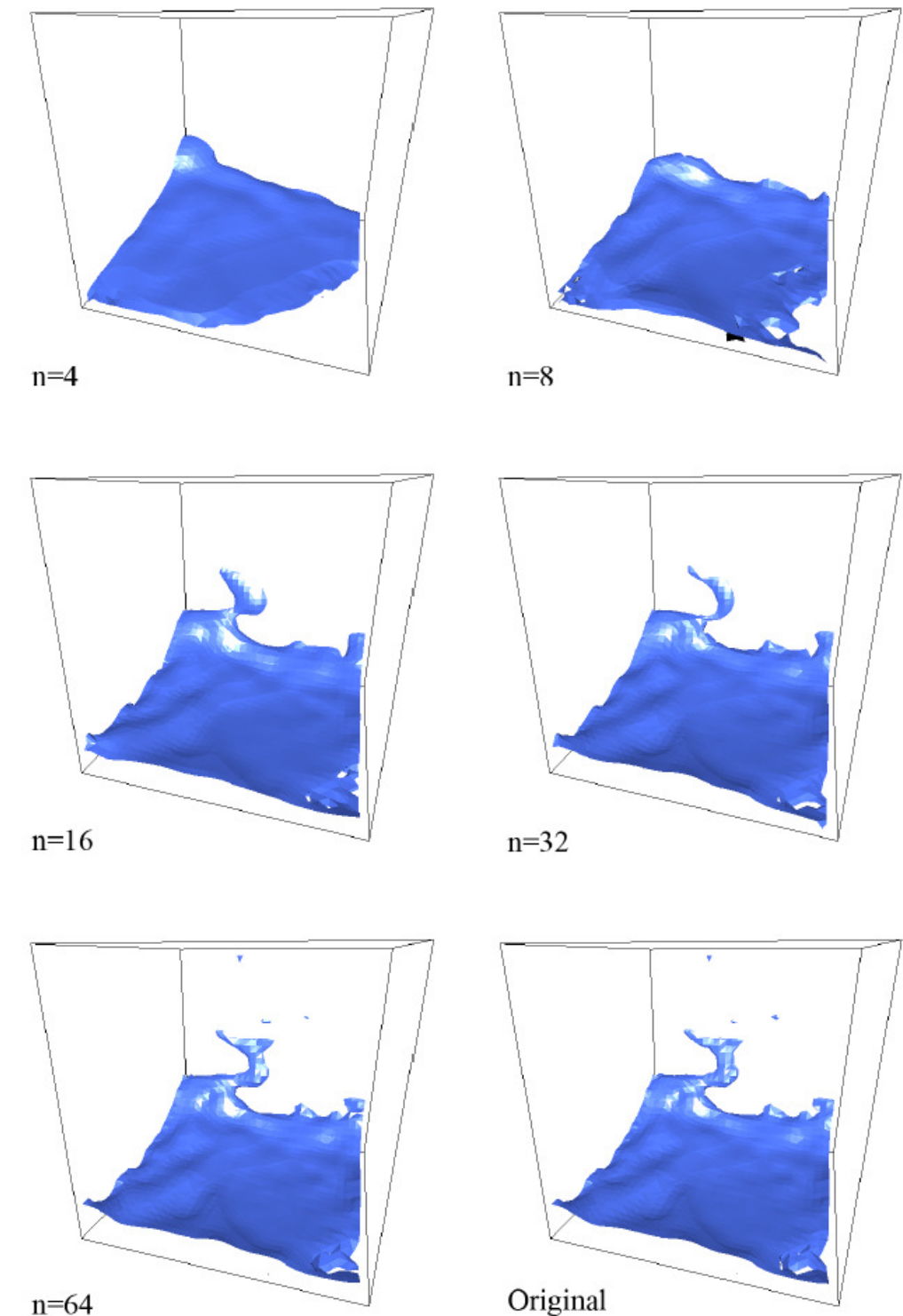
$$\tilde{\vec{x}}_j = \sum_{i=0}^{M-1} a_i \cdot \vec{b}_i \quad \Longrightarrow \quad \tilde{\vec{x}}_j \approx \sum_{i=0}^{M-1} y_i(t=j) \cdot \vec{b}_i$$

06 Ergebnisse

- Eigenwerte sind Maß für die Rekonstruktionsqualität
- In der Literatur findet sich ein Wert von 70 Prozent als zufriedenstellend [Ser02] - Praxis zeigt, dass auch niedrigere Werte visuell überzeugen können

Zeit- reihe	T	N ($\epsilon > 70\%$)	$K\epsilon$	N (visuell)	Kv
1	200	64	3,06	39	5,02
2	300	81	3,59	60	4,85
3	200	70	2,8	60	3,26
4	133	39	3,36	22	5,96
5	151	48	3,09	20	7,43
Ø	196,8	60,4	3,18	40,2	5,3

Kompressionsraten nach statischer Schwelle ($K\epsilon$) und visueller Beurteilung (Kv). T... Anz. Der Zeitschritte, N... Anz. der verwendeten Hauptkomponenten

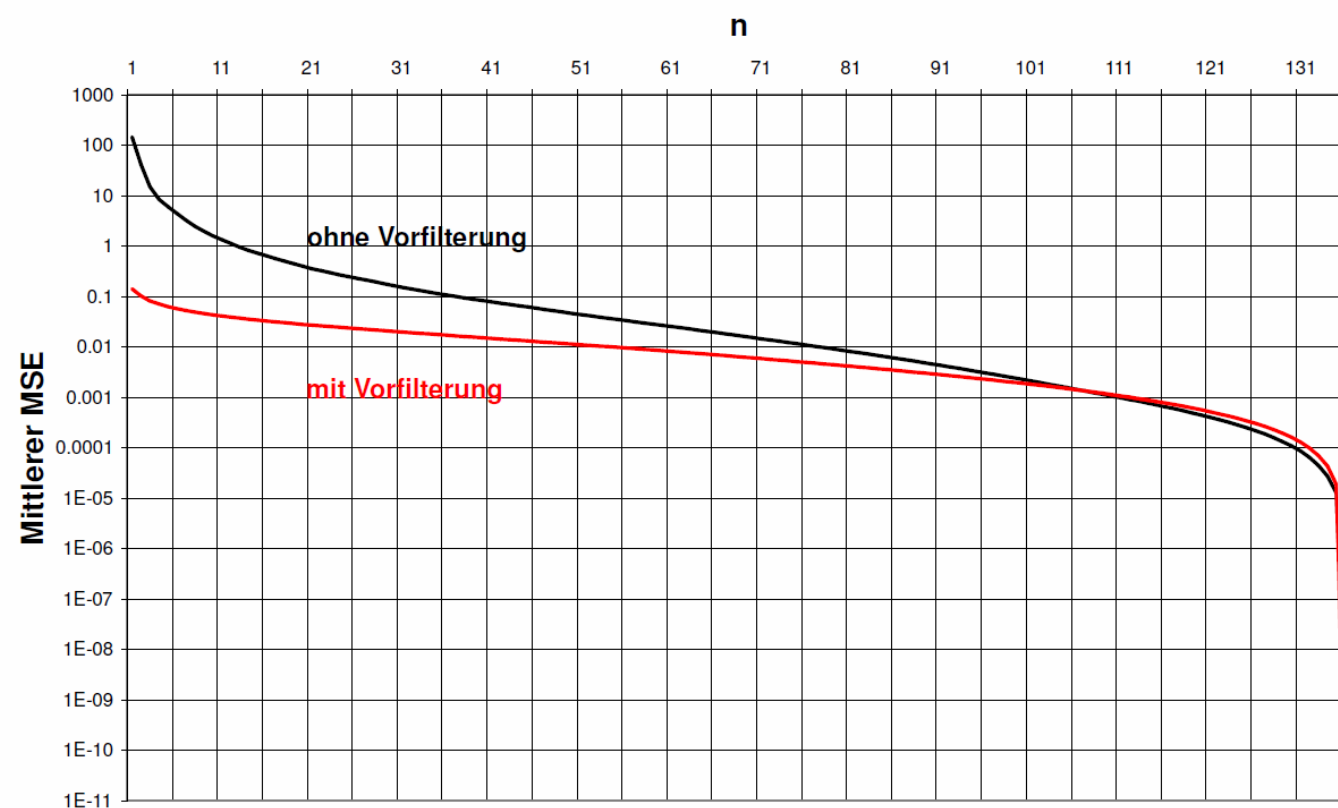


Rekonstruktionen eines 3D-Datensatzes mit unterschiedlicher Anzahl an Hauptkomponenten

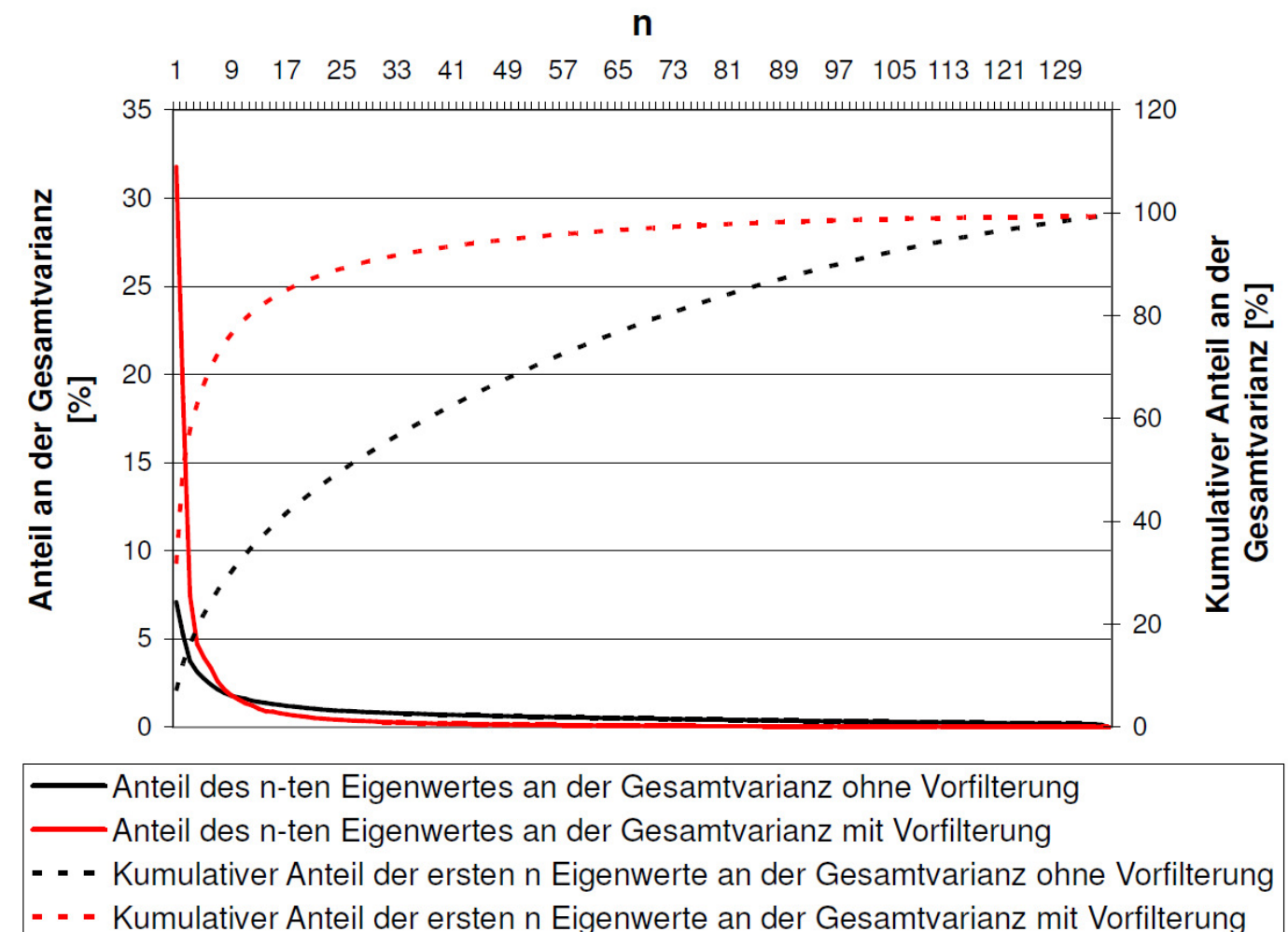
06 Ergebnisse

- Wichtig für das Gelingen war die Einführung eines Vorfilterungsschrittes:
- dadurch andere Verteilung der Varianzen und deutlich kleinere Fehler:

$$filt(\vec{x}_i, \varepsilon) \equiv \begin{cases} 1 & \text{wenn } \vec{x}_i > \varepsilon \\ -1 & \text{wenn } -\vec{x}_i > \varepsilon \\ 0 & \text{sonst} \end{cases}$$



Mittlerer MSE der Testzeitreihen (gemittelt über die Gesamtheit aller Zeitreihen) in Abhängigkeit der Vorfilterung sowie der Anzahl n der verwendeten Hauptkomponenten.



06 Ergebnisse

Simulation:

- einfache periodische Ausschwingvorgänge können animiert werden
- Simulationszeitraum unabhängig vom Trainingszeitraum
- Modellparameter zur Laufzeit einstellbar
- basierend auf dem vorhandenen Bildmaterial als Trainingsdaten, nicht physikalisch korrekt

07 Diskussion

PCA:

- effizient anwendbar durch Orthogonalität
 - schlägt jedoch fehl, wenn Richtungen der größten Varianz nicht zwangsläufig Richtungen der größten Bedeutsamkeit sind oder diese nicht orthogonal stehen → ICA [H000]
 - speicherhungrig, Kovarianzmatrix wächst quadratisch (128^3 -Set → 16 TiB bei 4 Byte float)
- SVD mit transponierter Datenmatrix (wenn $N \gg T$)
- schlechte Ergebnisse ohne Vorfilterung (in diesem Fall Optimierung des gesamten Distanzfeldes)

KPCA:

- Ergebnisse stehen noch aus
- Problem Mittelwertbefreiung und geeigneten Kern finden

Simulation:

- keine chaotischen Vorgänge simulierbar
- nicht interaktiv im Sinne der direkten Interaktion mit der Oberfläche, lediglich Parameter

08 Demonstration

Vielen Dank für die Aufmerksamkeit!

Quellen

- [Bak05] BAKER, Kirk: Singular Value Decomposition Tutorial. March 2005. – http://www.ling.ohio-state.edu/~kbaker/pubs/Singular_Value_Decomposition_Tutorial.pdf
- [HO00] HYVÄRINEN, Aapo ; OJA, Erkki: Independent component analysis: algorithms and applications. In: Neural Networks 13 (2000), S. 411–430
- [Jol02] JOLLIFFE, I.T.: Principal Components Analysis. Second Edition. Springer-Verlag New York, 2002 (Springer Series in Statistics). ISSN 0172–7397
- [Kut04] KUTRUFF, Heinrich: Akustik: Eine Einführung. S. Hirzel Verlag Stuttgart, April 2004. – ISBN 3777612448
- [OF02] OSHER, Stanley ; FEDKIW, Ronald: Level Set Methods and Dynamic Implicit Surfaces. Springer-Verlag New York, October 2002. – ISBN 0387954821
- [SA94] SREERAM, V. ; AGATHOKLIS, P.: On the properties of Gram matrix. In: Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on 41 (1994), Mar, Nr. 3, S. 234–237. – ISSN 1057–7122
- [Ser02] SERRE, Denis: Graduate Texts in Mathematics. Bd. 216: Matrices - Theory and Applications. Springer-Verlag New York, 2002. – ISBN 978-0-387-95460-8
- [Shl03] SHLENS, Jon: A Tutorial on Principal Components Analysis – Derivation, Discussion and Singular Value Decomposition. March 2003. – http://www.cs.princeton.edu/picasso/mats/PCA-Tutorial-Intuition_jp.pdf
- [SMS99] SCHÖLKOPF, Bernhard ; MÜLLER, Klaus-Robert ; SMOLA, Alexander J.: Lernen mit Kernen – Support-Vektor-Methoden zur Analyse hochdimensionaler Daten. In: Informatik – Forschung und Entwicklung 14 (1999), Nr. 3, S. 154–163. – ISSN 0178–3564
- [SS01] SCHÖLKOPF, Bernhard ; SMOLA, Alexander J.: Learning with Kernels – Support Vector Machines, Regularization, Optimization, and Beyond. The MIT Press, December 2001. – ISBN 0262194759
- [TLP06] TREUILLE, Adrien ; LEWIS, Andrew ; POPOVIĆ, Zoran: Model Reduction for Real-time Fluids. In: International Conference on Computer Graphics and Interactive Techniques 25 (2006), Nr. 3, S. 826–834. – ISSN 0730–0301
- [YZZL08] YAN, Su ; ZHAO, Jiu-Fen ; ZHAO, Jiu-Ling ; LI, Qing-Zhen: A Method for Image Classification based on Kernel PCA. In: Proceedings of the Seventh International Conference on Machine Learning and Cybernetics 2 (2008), July, S. 718–722. ISBN 978–1–4244–2095–7

Anhang – Herleitung KPCA

Kovarianzmatrix: $\hat{C} = 1/T \sum_{j=0}^{T-1} \vec{\Phi}(\vec{x}_j) \cdot \vec{\Phi}(\vec{x}_j)^T$ (1)

Eigenwertproblem: $\lambda \vec{b} = \hat{C} \vec{b}$

Alle Lösungen für \vec{b} liegen in linearer Hülle von $(\Phi(x_0) \dots \Phi(x_{T-1}))$, somit gilt:

$$\forall t = 0 \dots T-1: \langle \vec{\Phi}(\vec{x}_t), \lambda \vec{b} \rangle = \langle \vec{\Phi}(\vec{x}_t), \hat{C} \vec{b} \rangle \quad (2)$$

Zudem lassen sich die \vec{b} als Linearkombinationen der $\Phi(x_i)$ darstellen:

$$\vec{b} = \sum_{i=0}^{T-1} \alpha_i \vec{\Phi}(\vec{x}_i) \quad (3)$$

Anhang – Herleitung KPCA

(3) in (2):

$$\forall t = 0 \dots T-1: \left\langle \vec{\Phi}(\vec{x}_t), \lambda \sum_{i=0}^{T-1} \alpha_i \vec{\Phi}(\vec{x}_i) \right\rangle = \left\langle \vec{\Phi}(\vec{x}_t), \hat{C} \sum_{i=0}^{T-1} \alpha_i \vec{\Phi}(\vec{x}_i) \right\rangle \quad (4)$$

(1) in (4):

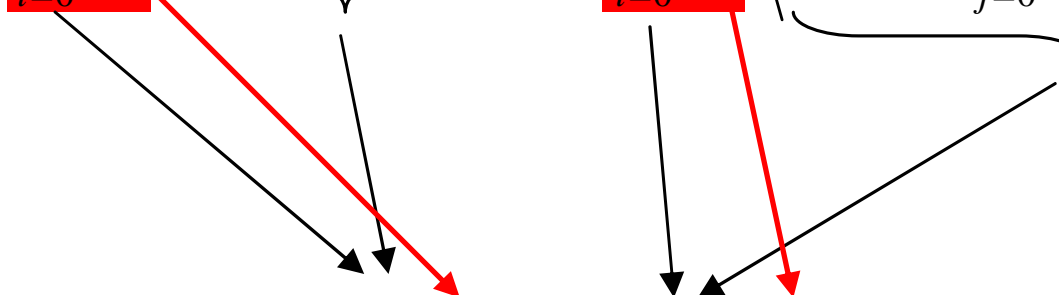
$$\forall t = 0 \dots T-1: \left\langle \vec{\Phi}(\vec{x}_t), \lambda \sum_{i=0}^{T-1} \alpha_i \vec{\Phi}(\vec{x}_i) \right\rangle = \left\langle \vec{\Phi}(\vec{x}_t), 1/T \sum_{j=0}^{T-1} \vec{\Phi}(\vec{x}_j) \cdot \vec{\Phi}(\vec{x}_j)^T \cdot \sum_{i=0}^{T-1} \alpha_i \vec{\Phi}(\vec{x}_i) \right\rangle$$

$$\lambda \sum_{i=0}^{T-1} \alpha_i \left\langle \vec{\Phi}(\vec{x}_t), \vec{\Phi}(\vec{x}_i) \right\rangle = 1/T \sum_{i=0}^{T-1} \alpha_i \left\langle \vec{\Phi}(\vec{x}_t), \sum_{j=0}^{T-1} \vec{\Phi}(\vec{x}_j) \left\langle \vec{\Phi}(\vec{x}_j), \vec{\Phi}(\vec{x}_i) \right\rangle \right\rangle$$

Anhang – Herleitung KPCA

$$\lambda \sum_{i=0}^{T-1} \alpha_i \langle \vec{\Phi}(\vec{x}_t), \vec{\Phi}(\vec{x}_i) \rangle = 1/T \sum_{i=0}^{T-1} \alpha_i \left\langle \vec{\Phi}(\vec{x}_t), \sum_{j=0}^{T-1} \vec{\Phi}(\vec{x}_j) \langle \vec{\Phi}(\vec{x}_j), \vec{\Phi}(\vec{x}_i) \rangle \right\rangle$$

$$T\lambda \sum_{i=0}^{T-1} \alpha_i \langle \vec{\Phi}(\vec{x}_t), \vec{\Phi}(\vec{x}_i) \rangle = \sum_{i=0}^{T-1} \alpha_i \left\langle \vec{\Phi}(\vec{x}_t), \sum_{j=0}^{T-1} \vec{\Phi}(\vec{x}_j) \langle \vec{\Phi}(\vec{x}_j), \vec{\Phi}(\vec{x}_i) \rangle \right\rangle$$



$$T\lambda K \vec{\alpha} = K^2 \vec{\alpha}$$

Anhang – Herleitung KPCA

$$T\lambda K\vec{\alpha} = K^2\vec{\alpha}$$

Die Lösungen von $\tilde{\lambda}\vec{\alpha} = K\vec{\alpha}$ mit $\tilde{\lambda} = T\lambda$ erfüllen das Eigenwertproblem.

Anschließende Normierung der $\vec{\alpha}$, sodass korrespondierende \vec{b} die Länge Eins haben,

d.h. $\forall i : \langle \vec{b}_i, \vec{b}_i \rangle = 1$

→ Normierungsvorschrift: mit $\vec{b} = \sum_{i=0}^{T-1} \alpha_i \vec{\Phi}(\vec{x}_i)$ ergibt sich $1 \equiv \langle \vec{\alpha}, K\vec{\alpha} \rangle$

$$1 \equiv \langle \vec{\alpha}, K\vec{\alpha} \rangle = \langle \vec{\alpha}, \lambda\vec{\alpha} \rangle = \lambda \langle \vec{\alpha}, \vec{\alpha} \rangle \Rightarrow \tilde{\alpha} = \frac{\vec{\alpha}}{\sqrt{\tilde{\lambda}}} \quad \text{für } \lambda > 0$$

Anhang – Herleitung KPCA

Merkmalsextraktion bei M Hauptkomponenten:

$$a_{t,i} \equiv \langle \vec{\Phi}(\vec{x}_t), \vec{b}_i \rangle = \sum_{m=0}^{M-1} \tilde{\alpha}_{i,m} \langle \vec{\Phi}(\vec{x}_t), \vec{\Phi}(\vec{x}_m) \rangle \quad \text{mit } \vec{b} = \sum_{i=0}^{T-1} \alpha_i \vec{\Phi}(\vec{x}_i)$$

Unter Zuhilfenahme der Kernfunktionen k sowie Kernmatrix K ergibt sich:

$$a_{t,i} \equiv \sum_{m=0}^{M-1} \tilde{\alpha}_{i,m} k(\vec{x}_t, \vec{x}_m) = \sum_{m=0}^{M-1} \tilde{\alpha}_{i,m} K_{mt} = \langle \vec{\tilde{\alpha}}_i, \vec{k}_t \rangle \quad \text{mit } \vec{k}_t \text{ als t-te Spalte von } K$$

Matrix aller Merkmale:

$$A = (\vec{a}_0 \dots \vec{a}_{T-1}) = \begin{pmatrix} \tilde{\alpha}_0 \\ \vdots \\ \tilde{\alpha}_{M-1} \end{pmatrix} \cdot K$$