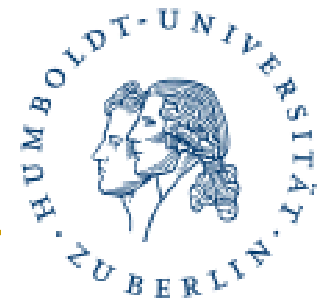


Inpainting



Jane Dienemann

Benjamin Vorwerk

Seminar Computational Photography

Dozent: Prof. Dr.-Ing. Eisert

Sommersemester 2010

Übersicht

I. Der Begriff „Inpainting“

II. Die Geschichte der Bildbearbeitung

1) Gemälde, Photographie

2) Digitale Photographie

III. Aktuelle Methoden der Bildbearbeitung

1) Vorstellung verschiedener Ansätze und Probleme

2) Vertiefung eines Algorithmus: Stärken, Schwächen

IV. Aussichten

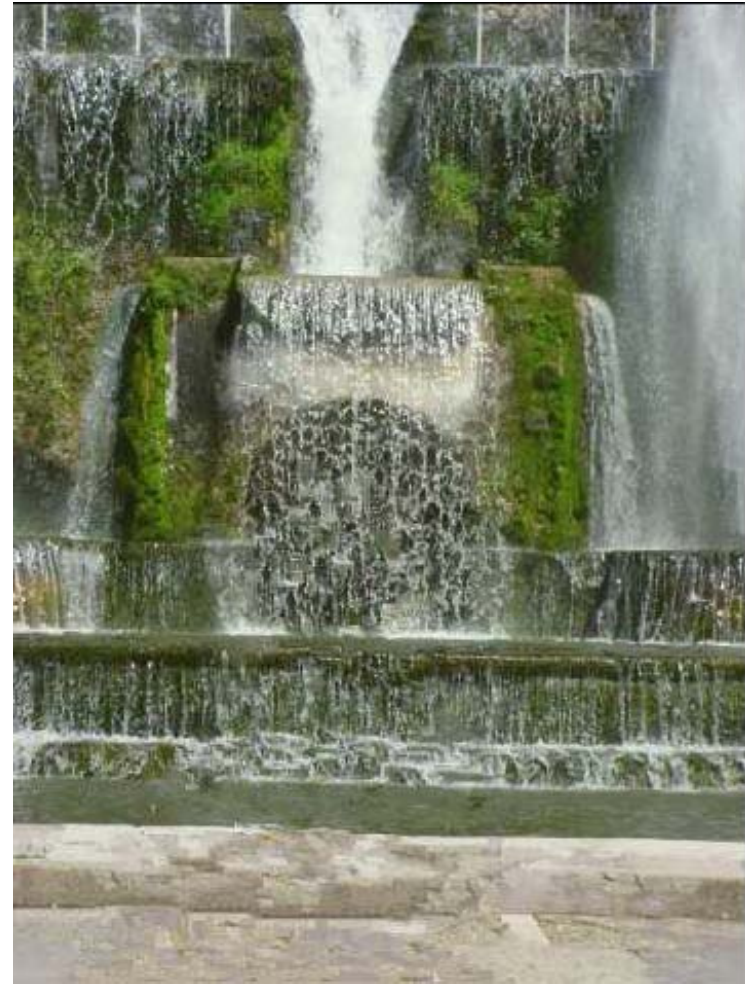
V. Fragen / Diskussion

I. Der Begriff „Inpainting“

- - Wiederherstellung von zerstörten oder nicht mehr vorhanden Teilen eines Bildes oder Videos
- - im digitalen Bereich: Verwendung von komplexen Algorithmen



Rekonstruktion von Bildern



Entfernen von Objekten (großflächig)

I. Der Begriff „Inpainting“



Entfernen von Objekten (feinere Strukturen)

I. Der Begriff „Inpainting“



Hinzufügen von Farbinformationen



initial image



restored image

Entfernen von Texten



nahtloses Einfügen von Objekten



nahtloses Einfügen von Objekten

II. Die Geschichte der Bildbearbeitung

1) *Gemälde, Photographie*

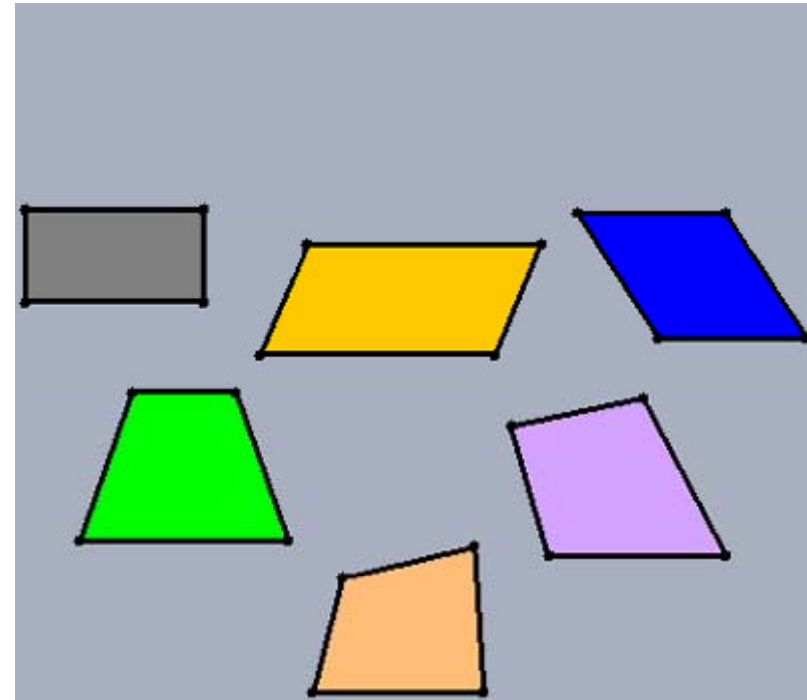
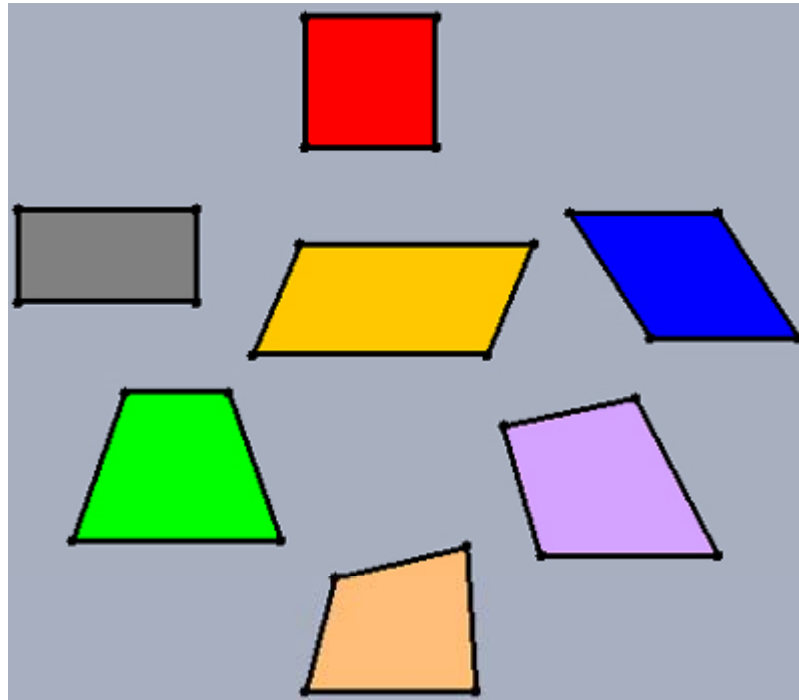


Rekonstruktion eines Gemäldes

Schabretusche
(*analoges Bild*)



2) Digitale Photographie

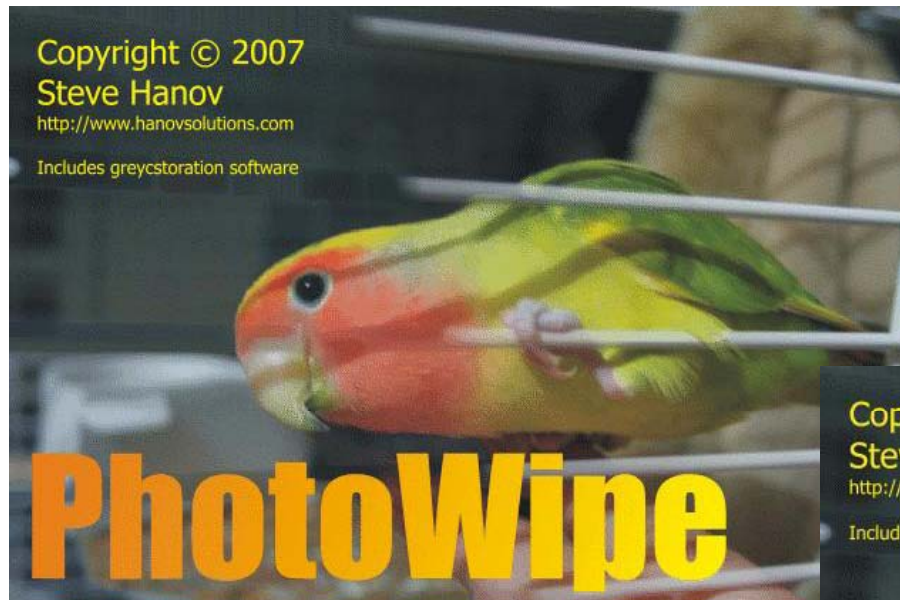


simple Bearbeitung mit Photoshop (Kopierstempel)

- Objekt wird markiert und mit Hintergrundfarbe ersetzt

III. Aktuelle Methoden der Bildbearbeitung

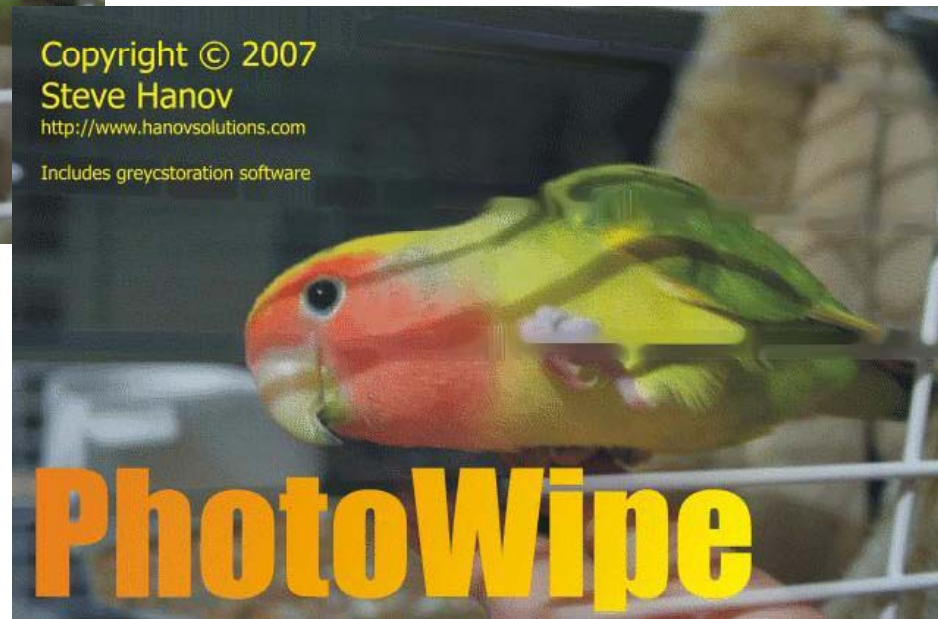
1) Vorstellung verschiedener Ansätze und Probleme



Wunschvorstellung des
Herstellers

tatsächliches Resultat

**Beispiel für schlechte
Inpainting Software**

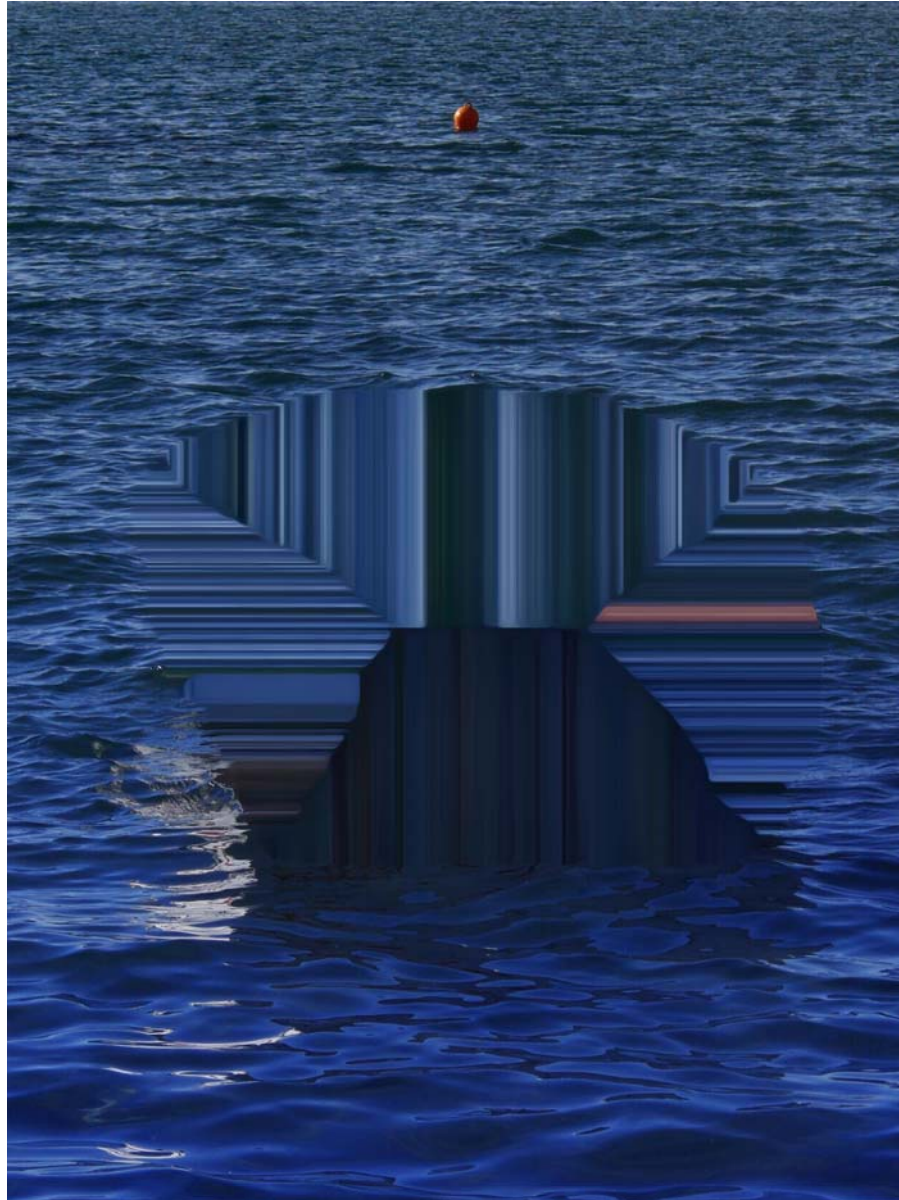


noch ein Beispiel

Originalbild

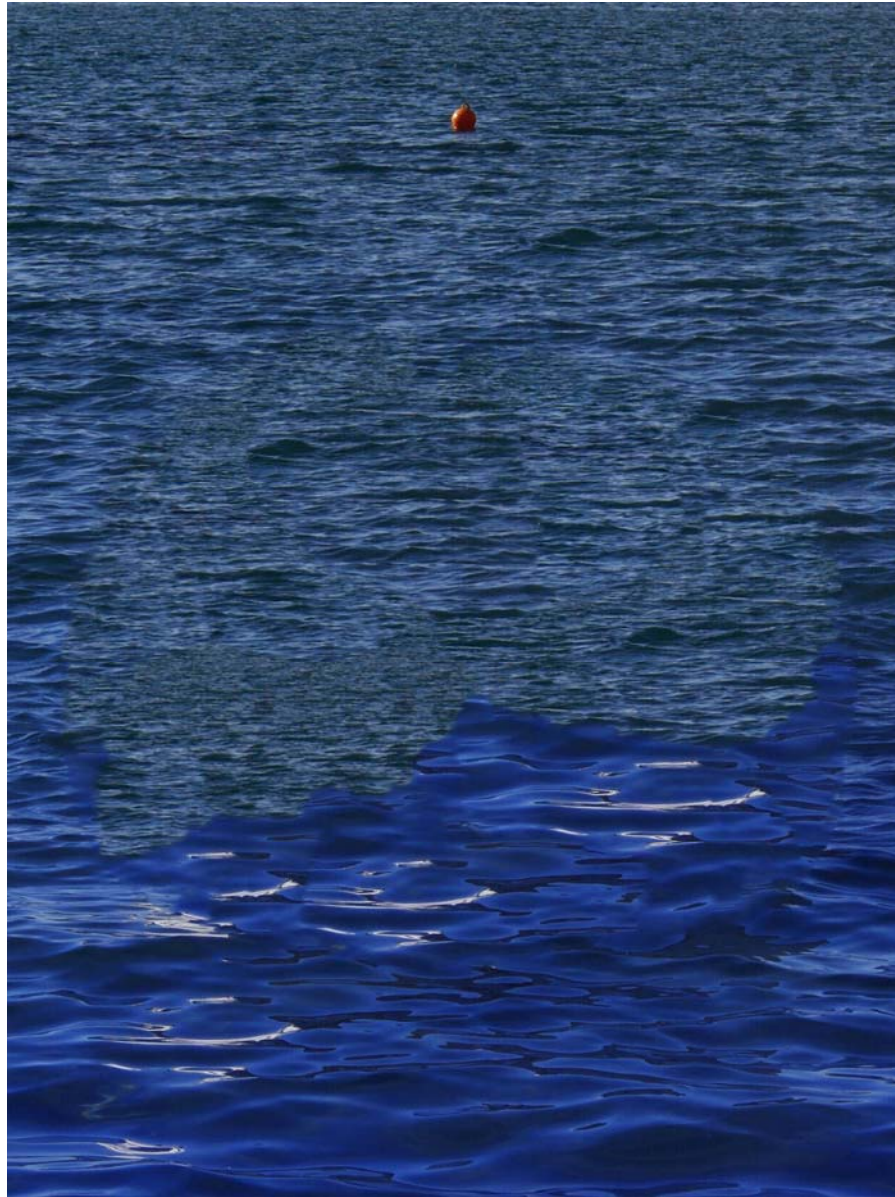


bearbeitet
(mit Photo Wipe)



bearbeitet
(mit Photoshop CS5)

im Vergleich:
sehr gutes Ergebnis



bearbeitet
(mit Photoshop CS5)



Noch ein Beispiel (mit Photoshop CS5)

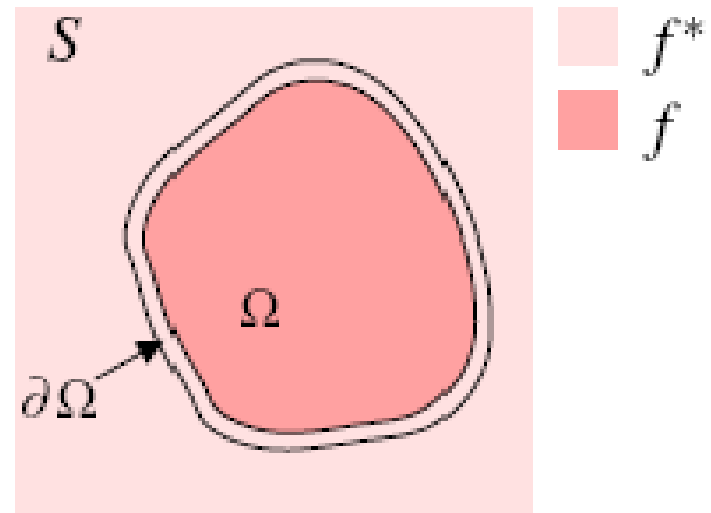
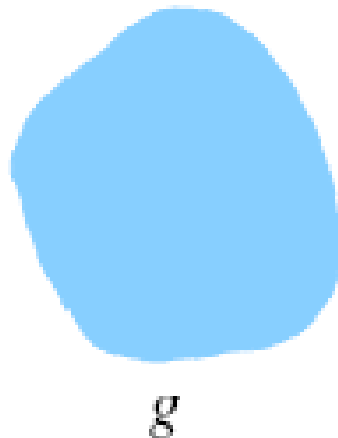
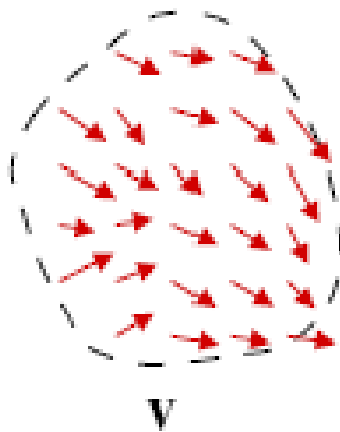


Noch ein Beispiel (mit Photoshop CS5)

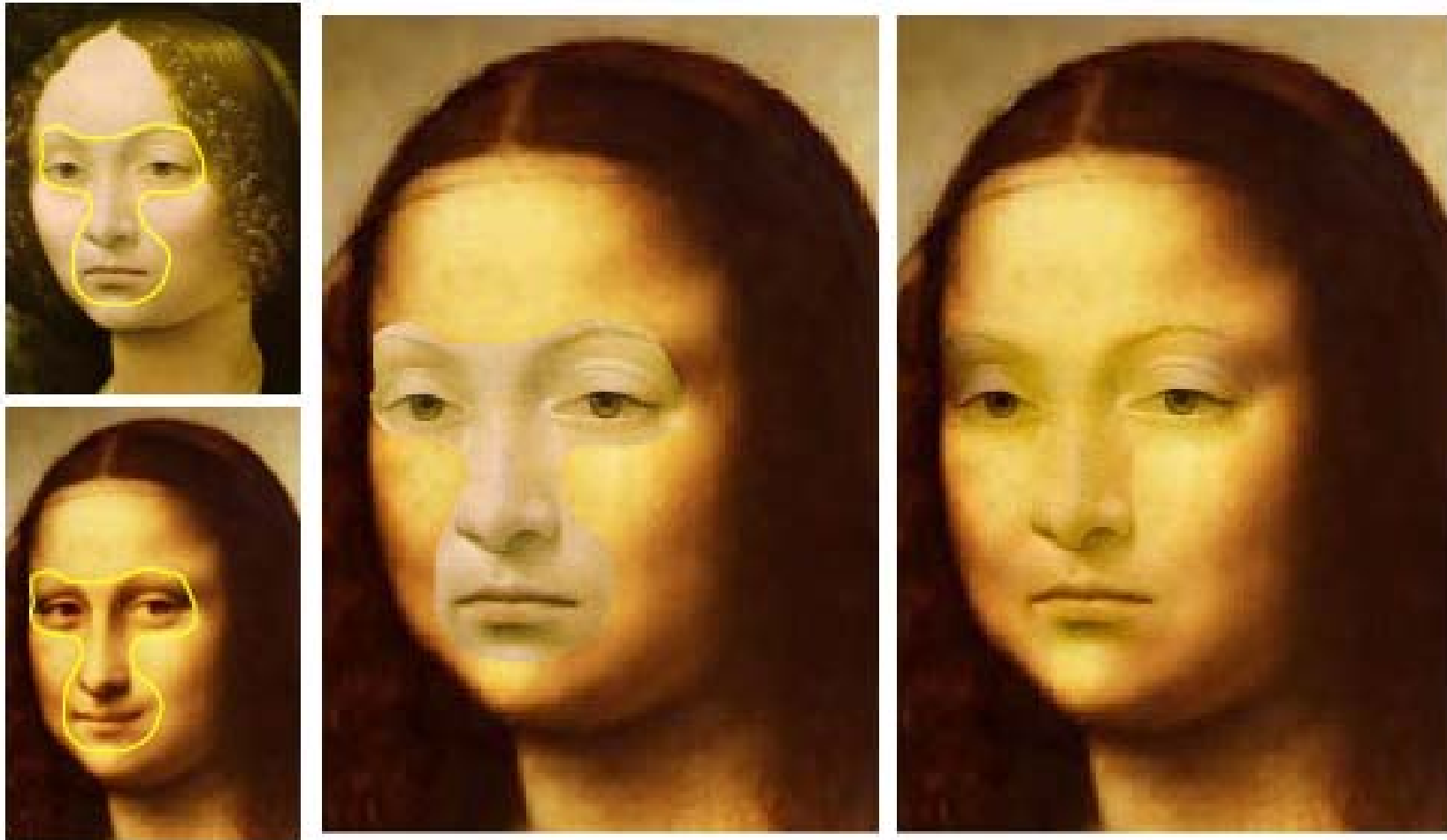


1) Vorstellung verschiedener Ansätze und Probleme

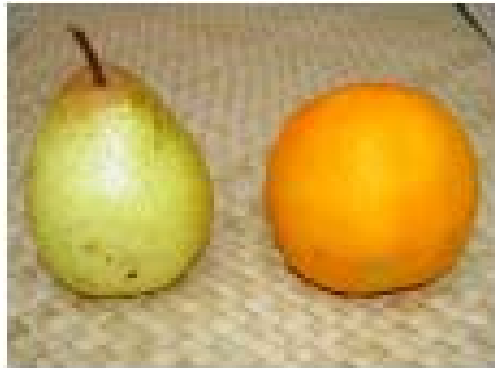
Gradientenvektorfelder und partielle Differenzialgleichungen:



Beispiel:



Beispiel:





III. Aktuelle Methoden der Bildbearbeitung

Ein Algorithmus zum Auffüllen großer Teilbereiche eines Bildes: *Anforderungen:*

- Strukturen, Kanten, etc. sollen fortgesetzt werden.
- Texturen sollen realistisch wirken.
- Eine Nachbearbeitung soll möglichst nicht auffallen.
- Der Algorithmus soll effizient und schnell sein.

„Region Filling and Object Removal by Exemplar-Based Image Inpainting“

Microsoft Research 2004.

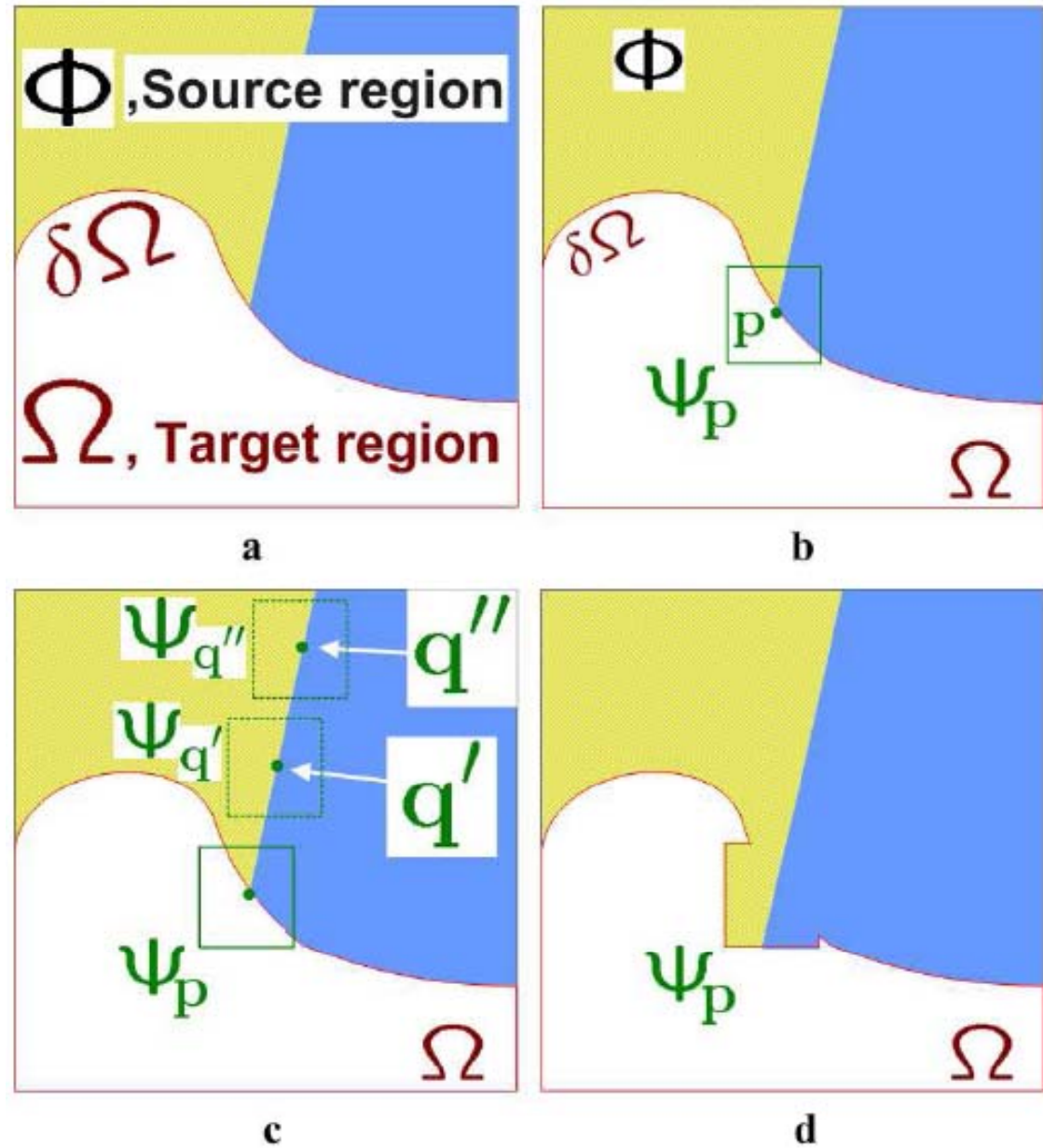
Eigenschaften:

- Gleichzeitige Synthese von Texturen und linearen Strukturen.
- Vollautomatische Bestimmung der Füllreihenfolge (wichtig).
- Effektive, schnelle Berechnung des neuen Bildes.
- Keine Unschärfe – Effekte (Verwischen).
- Keine unerwünschten Artefakte (Overshooting).

Ergebnis:

Ein optisch plausibel erscheinendes neues Bild.

Der Algorithmus:

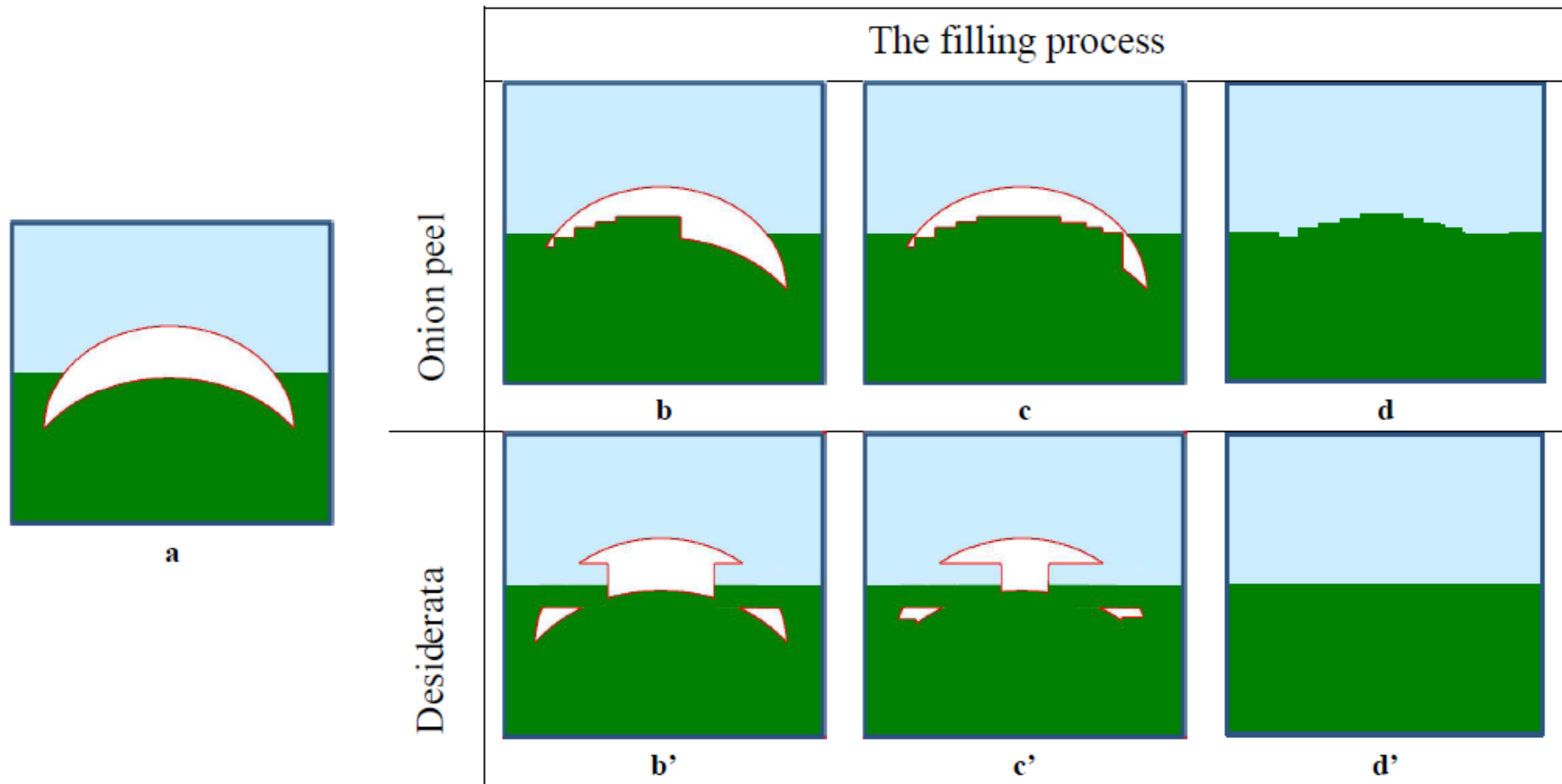


Die Wahl des ähnlichsten Quellstückes:

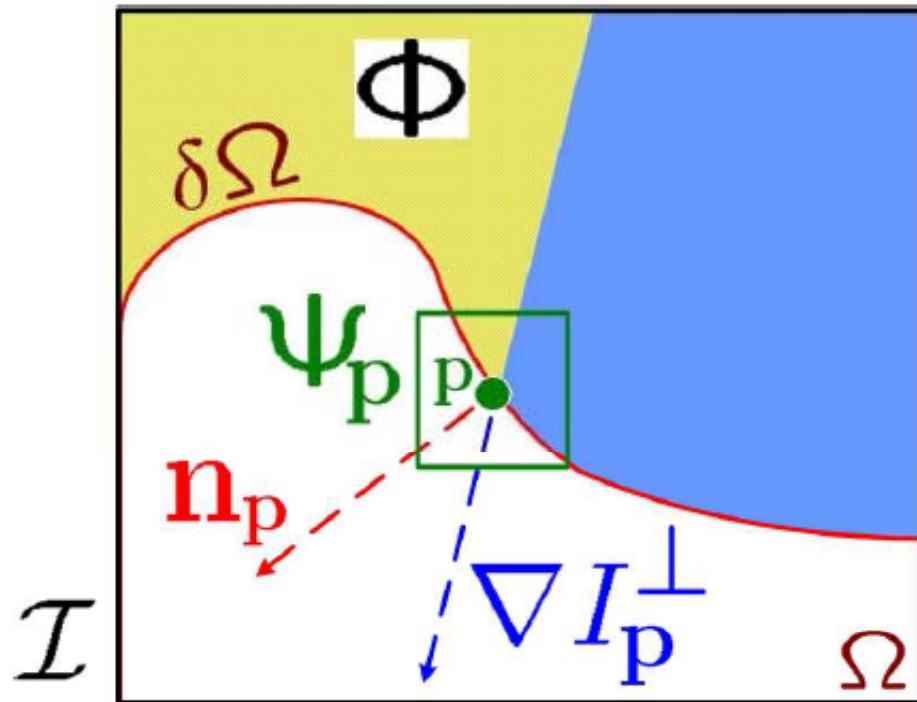
$$\Psi_{\hat{q}} = \arg \min_{\Psi_q \in \Phi} d(\Psi_{\hat{p}}, \Psi_q)$$

- Verwendeter Farbraum CIE Lab.
- Euklidischer Abstand im RGB Farbraum wenig aussagekräftig!

Die Reihenfolge des Auffüllens ist wichtig!



Die Bestimmung der Füllreihenfolge:



Priorität des Grenzpunktes p :

$$P(p) = C(p)D(p).$$

Wobei

$$C(p) = \frac{\sum_{q \in \Psi_p \cap (\mathcal{I} - \Omega)} C(q)}{|\Psi_p|}$$

und

$$D(p) = \frac{|\nabla I_p^\perp \cdot \mathbf{n}_p|}{\alpha}$$

Pseudocode des Algorithmus:

- Speichere die manuell gewählte initiale Füllfront $\delta\Omega^0$
- Wiederhole bis alles gefüllt ist:
 - Identifiziere die Füllfront $\delta\Omega^t$. Wenn $\delta\Omega^t = \{\}$, beende.
 - Berechne Prioritäten $P(p) \forall p \in \delta\Omega^t$.
 - Finde das Feld Ψp^h mit der höchsten Priorität.
 - Finde das ähnlichste Feld Ψp^a aus Φ .
 - Kopiere Bilddaten von Ψp^a nach $\Psi p^h \forall p \in \Psi p^h \cap \Omega$.
 - Aktualisiere Vertrauenswert $C(p) \forall p \in \Psi p^h \cap \Omega$.

Einige Beispiele und Vergleiche:



a



b



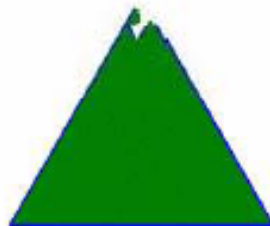
c



d



e



f

Einige Beispiele und Vergleiche:



a



b



c



d



e



f

Einige Beispiele und Vergleiche:



Einige Beispiele und Vergleiche:



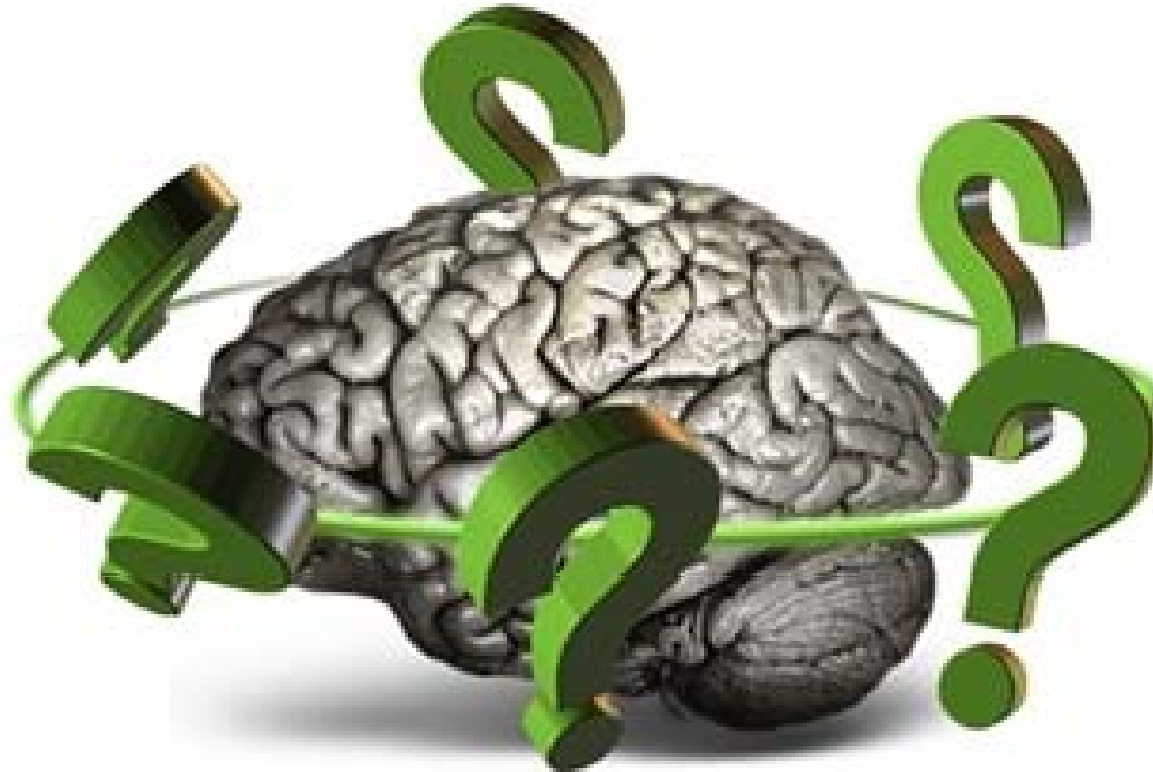
Schwächen des Algorithmus:

1. Füllen von Regionen für die es keine ähnlichen Texturteile gibt, oder die ein unrealistisches Ergebnis produzieren.
2. Der Algorithmus hat Schwierigkeiten abgerundete Strukturen zu ergänzen.
3. Tiefe wird nicht berücksichtigt (Welches Bildelement liegt vorne, welches hinten etc.).

IV. Aussichten

- Adobe Photoshop CS5. Funktion: „Content Aware“?!
 - Berücksichtigung von Tiefe.
 - Vervollständigung nicht-linearer Strukturen.
 - Echtzeit-Berechnung in Filmen, bzw. bei Live-Übertragungen.
-

V. Fragen / Diskussion



Quellen

- A. Criminisi, P. Pérez, K. Toyama. Region Filling and Object Removal by Exemplar-Based Image Inpainting. IEEE Transactions on Image Processing, Vol.13, No.9, Sep. 2004.
 - P. Pérez, M. Gangnet, A. Blake. Poisson Image Editing. Microsoft Research UK. 2003
 - Nikos Komodakis, Georgios Tziritas. Image Completion Using Efficient Belief Propagation Via Priority Scheduling and Dynamic Pruning. IEEE Transactions on Image Processing, Vol.16, No.11, Nov. 2007.
 - Guillermo Sapiro. Inpainting the Colors. INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS Preprint Series #1979. May 2004.
 - Patrick Pérez, Michel Gangnet, Andrew Blake. PatchWorks: Example-Based Region Tiling for Image Editing. Microsoft Research US. January 13, 2004.
 - Chenyu Wu, Ce Liu, Heung-Yeung Shum, Ying-Qing Xu, Zhengyou Zhang. Automatic Eyeglasses Removal from Face Images. ACCV2002: The 5th Asian Conference on Computer Vision, 23–25 January 2002, Melbourne, Australia.
-