Seam Carving

Sebastian Arzt, Tim Rocktäschel

Humboldt Universität zu Berlin
Seminar Computational Photography
Sommersemester 2010

19. Juni 2010

Motivation

- Problem:
  - Heutzutage: vielfältige Displaytypen
  - Unterschiede in Auflösung und Seitenverhältnis
  - z.B. Bilder/Filme auf 16:10 LCD, Handydisplay, altem 4:3 Monitor
  - HTML kann sich dynamisch anpassen, aber Bilder nicht

Motivation

- Problem:
  - Heutzutage: vielfältige Displaytypen
  - Unterschiede in Auflösung und Seitenverhältnis
  - z.B. Bilder/Filme auf 16:10 LCD, Handydisplay, altem 4:3 Monitor
  - HTML kann sich dynamisch anpassen, aber Bilder nicht
- Bisher:
  - Scaling $\rightarrow$ gestauchte/gestreckte Bilder
  - Cropping $\rightarrow$ Informationsverlust

Motivation

- Problem:
  - Heutzutage: vielfältige Displaytypen
  - Unterschiede in Auflösung und Seitenverhältnis
  - z.B. Bilder/Filme auf 16:10 LCD, Handydisplay, altem 4:3 Monitor
  - HTML kann sich dynamisch anpassen, aber Bilder nicht
- Bisher:
  - Scaling → gestauchte/gestreckte Bilder
  - Cropping → Informationsverlust
- Lösung: Content Aware Resizing
  - Anpassung von Bildern an vielfältige Auflösung und Seitenverhältnisse von heutigen Displays
  - Bildinhalte von Interesse dabei möglichst wenig verändern

→ weniger gestaucht/gestreckte Bilder mit geringerem Informationsverlust

Originalbild

Cropped

Scaled

Seam Carved

Vergleich

Gliederung

Formalisierung unwichtiger Bildinhalte

- Sei eine Energiefunktion $e : \mathbf{I} \rightarrow \mathbb{R}_+$ gegeben
- Ordnet jedem Pixel einen Energiewert zu
- Hohe Energie bedeutet hohe Wichtigkeit
- Naiv: entferne pro Zeite die k unwichtigsten Pixel

Formalisierung unwichtiger Bildinhalte

- Sei eine Energiefunktion $e : \mathbf{I} \to \mathbb{R}_+$ gegeben
- Ordnet jedem Pixel einen Energiewert zu
- Hohe Energie bedeutet hohe Wichtigkeit
- Naiv: entferne pro Zeite die k unwichtigsten Pixel

Formalisierung unwichtiger Bildinhalte

- Sei eine Energiefunktion $e : \mathbf{I} \to \mathbb{R}_+$ gegeben
- Ordnet jedem Pixel einen Energiewert zu
- Hohe Energie bedeutet hohe Wichtigkeit
- Naiv: entferne pro Zeite die k unwichtigsten Pixel

Beispiel: Energiematrix

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Definition Seam

- $\mathbf{I}$ ist ein $m \times n$ Bild
- Ein Seam ist ein Pixelpfad zwischen gegenüberliegenden Bildrändern

Definition Seam (cont.)

- Ein vertikaler Seam ist formal wie folgt definiert:

$$s^x = \{s_i{}^x\}_{i=1}^n = \{x(i), i\}_{i=1}^n$$

- $x(i)$ ordnet jeder Zeile genau ein Pixel zu

Definition Seam (cont.)

- Ein vertikaler Seam ist formal wie folgt definiert:

$$s^x = \{s_i{}^x\}_{i=1}^n = \{x(i), i\}_{i=1}^n$$

- $x(i)$ ordnet jeder Zeile genau ein Pixel zu
- Wichtige Bedingung:

$$\forall x : |x(i) - x(i-1)| \leq 1 \quad \text{Sinn?}$$

Definition Seam (cont.)

- Ein vertikaler Seam ist formal wie folgt definiert:

$$s^x = \{s_i{}^x\}_{i=1}^n = \{x(i), i\}_{i=1}^n$$

- $x(i)$ ordnet jeder Zeile genau ein Pixel zu
- Wichtige Bedingung:

$$\forall x : |x(i) - x(i-1)| \leq 1 \quad \text{Sinn?}$$

- Entfernen eines Seams erzeugt weniger sichtbare Verzerrungen
- Ab jetzt: horizontale Auflösung verändern $\rightarrow$ vertikale Seams
- Horizontale Seams analog

Seam Carving

- Gesucht ist Seam mit niedrigster Energie

$$s^* = \arg\min_s E(s) = \arg\min_s \sum_{i=1}^{n} e(\mathbf{I}(s_i))$$

Seam Carving

- Gesucht ist Seam mit niedrigster Energie

$$s^* = \arg\min_s E(s) = \arg\min_s \sum_{i=1}^{n} e(\mathbf{I}(s_i))$$

- Komplexität des Suchraums:
  - Wieviele vertikale Seams s in einem $m * n$ Bild gibt es?

Seam Carving

- Gesucht ist Seam mit niedrigster Energie

$$s^* = \arg\min_s E(s) = \arg\min_s \sum_{i=1}^{n} e(\mathbf{I}(s_i))$$

- Komplexität des Suchraums:
  - Wieviele vertikale Seams s in einem $m * n$ Bild gibt es?

$$O(2^n * m) \leq \#\text{Seams} \leq O(3^n * m)$$

Berechnung der Energie eines Seams

Sei $M(i, j)$ der Energiewert desjenigen vertikalen Seams

- welcher am oberen Bildrand beginnt,
- im Pixel (i, j) endet und
- die niedrigste Energie besitzt

Berechnung der Energie eines Seams

Sei $M(i, j)$ der Energiewert desjenigen vertikalen Seams

- welcher am oberen Bildrand beginnt,
- im Pixel (i, j) endet und
- die niedrigste Energie besitzt

Berechnung von $M(i, j)$:

M:	1	2	3	4	5
1					
2					
3					
4		?			
5					

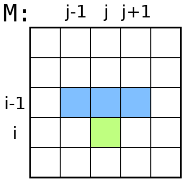
E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Rekursionsgleichung für $M(i, j)$

$$M(1, j) = e(1, j)$$

$$M(i, j) = e(i, j) + \min \begin{cases} M(i-1, j-1) \\ M(i-1, j) \\ M(i-1, j+1) \end{cases}$$

$$M(i, j) = \infty \quad \text{falls } i \notin [1, n] \text{ oder } j \notin [1, m]$$

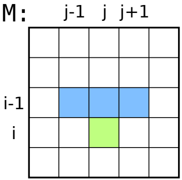


Rekursionsgleichung für $M(i,j)$

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min \begin{cases} M(i-1, j-1) \\ M(i-1, j) \\ M(i-1, j+1) \end{cases}$$

$$M(i,j) = \infty \quad \text{falls } i \notin [1,n] \text{ oder } j \notin [1,m]$$



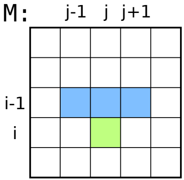
- Komplexität?

Rekursionsgleichung für $M(i,j)$

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min \begin{cases} M(i-1, j-1) \\ M(i-1, j) \\ M(i-1, j+1) \end{cases}$$

$$M(i,j) = \infty \quad \text{falls } i \notin [1,n] \text{ oder } j \notin [1,m]$$



- Komplexität? $O(3^i)$

Beobachtung

$$M(4,3)\begin{cases} M(3,2)\begin{cases} M(2,1)\\ M(2,2)\\ \textcolor{blue}{M(2,3)}\end{cases}\\ M(3,3)\begin{cases} M(2,2)\\ \textcolor{blue}{M(2,3)}\\ M(2,4)\end{cases}\\ M(3,4)\begin{cases} \textcolor{blue}{M(2,3)}\\ M(2,4)\\ M(2,5)\end{cases}\end{cases} \qquad M(4,4)\begin{cases} M(3,3)\begin{cases} M(2,2)\\ \textcolor{blue}{M(2,3)}\\ M(2,4)\end{cases}\\ M(3,4)\begin{cases} \textcolor{blue}{M(2,3)}\\ M(2,4)\\ M(2,5)\end{cases}\\ M(3,5)\begin{cases} M(2,4)\\ M(2,5)\\ M(2,6)\end{cases}\end{cases}$$

Beobachtung

$$M(4,3)\begin{cases} M(3,2)\begin{cases} M(2,1)\\ M(2,2)\\ {\color{blue}M(2,3)} \end{cases}\\ M(3,3)\begin{cases} M(2,2)\\ {\color{blue}M(2,3)}\\ M(2,4) \end{cases}\\ M(3,4)\begin{cases} {\color{blue}M(2,3)}\\ M(2,4)\\ M(2,5) \end{cases} \end{cases} \qquad M(4,4)\begin{cases} M(3,3)\begin{cases} M(2,2)\\ {\color{blue}M(2,3)}\\ M(2,4) \end{cases}\\ M(3,4)\begin{cases} {\color{blue}M(2,3)}\\ M(2,4)\\ M(2,5) \end{cases}\\ M(3,5)\begin{cases} M(2,4)\\ M(2,5)\\ M(2,6) \end{cases} \end{cases}$$

Tatsächliche Komplexität des Problems ist $O(m * n)$!

Dynamische Programmierung

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min\left\{M(i-1,j-1), M(i-1,j), M(i-1,j+1)\right\}$$

M:	1	2	3	4	5
1					
2					
3					
4					
5					

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1,j) = e(1,j)$$
$$M(i,j) = e(i,j) + \min\left\{M(i-1,j-1), M(i-1,j), M(i-1,j+1)\right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2					
3					
4					
5					

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min\left\{M(i-1,j-1), M(i-1,j), M(i-1,j+1)\right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2	4				
3					
4					
5					

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min \left\{ M(i-1,j-1), M(i-1,j), M(i-1,j+1) \right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2	4	3			
3					
4					
5					

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min \left\{ M(i-1,j-1), M(i-1,j), M(i-1,j+1) \right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2	4	3	7		
3					
4					
5					

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min\left\{M(i-1,j-1), M(i-1,j), M(i-1,j+1)\right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2	4	3	7	5	
3					
4					
5					

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min \left\{ M(i-1,j-1), M(i-1,j), M(i-1,j+1) \right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2	4	3	7	5	9
3					
4					
5					

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min\left\{M(i-1,j-1), M(i-1,j), M(i-1,j+1)\right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2	4	3	7	5	9
3	12	5	7	13	9
4					
5					

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1, j) = e(1, j)$$

$$M(i, j) = e(i, j) + \min\left\{M(i-1, j-1), M(i-1, j), M(i-1, j+1)\right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2	4	3	7	5	9
3	12	5	7	13	9
4	12	11	9	16	15
5					

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1, j) = e(1, j)$$
$$M(i, j) = e(i, j) + \min \left\{ M(i-1, j-1), M(i-1, j), M(i-1, j+1) \right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2	4	3	7	5	9
3	12	5	7	13	9
4	12	11	9	16	15
5	19	11	17	16	24

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

Dynamische Programmierung

$$M(1,j) = e(1,j)$$

$$M(i,j) = e(i,j) + \min\left\{M(i-1,j-1), M(i-1,j), M(i-1,j+1)\right\}$$

M:	1	2	3	4	5
1	1	2	0	4	3
2	4	3	7	5	9
3	12	5	7	13	9
4	12	11	9	16	15
5	19	11	17	16	24

E:	1	2	3	4	5
1	1	2	0	4	3
2	3	3	7	5	6
3	9	2	4	8	4
4	7	6	4	9	6
5	8	2	8	7	9

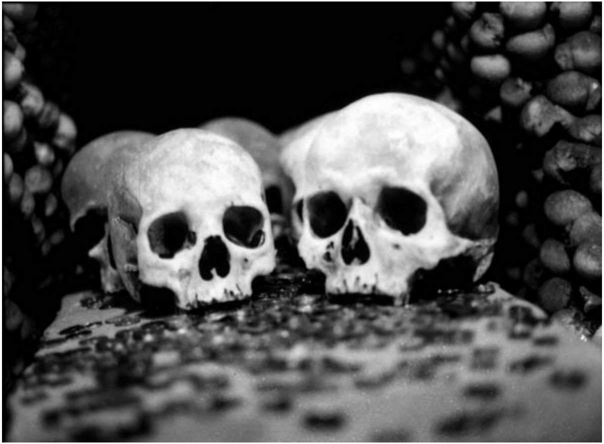
Content Aware Resizing

- Entfernen von horizontalen und vertikalen Seams
- Optimale Reihenfolge über dynamische Programmierung bestimmbar

Content Amplification

Content Amplification

Content Amplification



Content Amplification

Content Amplification

Content Amplification

Content Amplification

Object Removal

Object Removal

TODO

maximal 1 Folie

TODO

Grundideen auf maximal 2 Folien

- *http://liquidrescale.wikidot.com/, 19.06.2010*
- *Avidan, S. and Shamir, A., Seam carving for content-aware image resizing, ACM Transactions on Graphics, 2007*
- *Rubinstein, M. and Shamir, A. and Avidan, S., Improved seam carving for video retargeting, ACM Transactions on Graphics-TOG, 2008*