Decomposing Time-Lapse Paintings into Layers

Jianchao Tan  George Mason University
Marek Dvorožňák  Czech Technical University in Prague
Daniel Sýkora  Czech Technical University in Prague
Yotam Gingold  George Mason University
Background: Digital Painting

Background: Digital Painting

Background: Digital Painting

Layers are RGBA images

Background: Digital Painting

Motivation

- Physical paintings are hard to edit.
Motivation

• What if we have a time lapse video?
Motivation

• What if we have a time lapse video?
Goal

- Decompose a time-lapse painting video into layers
Goal

• Decompose a time-lapse painting video into layers
Goal

- Decompose a time-lapse painting video into layers
Challenges

• Preprocessing:
Challenges

• Preprocessing:

painter
Challenges

- Preprocessing:

  painter

  shadows
Challenges

- Preprocessing:
  - color shift
  - painter
  - shadows
Challenges

• Preprocessing:
  - color shift
  - lighting
  - painter
  - shadows
Challenges

- Recovering paint layers
Challenges

• Recovering paint layers

before

after
Related Work

- Interacting with editing history
  - Su et al. [2009], VisTrails [2009], McCann and Pollard [2009; 2012], Grossman et al. [2010], Noris et al. [2012], Denning and Pellacini [2013], Chen et al. [2014], Matzen and Snavely [2014], Karsch et al. [2014].

Chronicle [Grossman et al. 2010]
Related Work

• Decomposing edits
  • Xu et al. [2006], Amati and Brostow [2010], Fu et al. [2011], Hu et al. [2013], Richardt et al. [2014].
Related Work

- Image matting
  - Smith and Blinn [1996], Zongker et al. [1999], Farid and Adelson [1999], Szeliski et al. [2000], Levin et al. [2006; 2007]

Blue Screen Matting [Smith and Blinn 1996]
Pipeline

Input
Preprocess
Extract Layers
Edit
Pipeline

Input

Preprocess

Extract Layers

Edit
Pipeline

Time lapse recording

Input
Pipeline

Time lapse recording

Input
Pipeline

Preprocess
Pipeline
Pipeline

Input

Extract Layers
Pipeline

Input

Preprocess

Extract Layers

Edit
Pipeline

Interactive editing using our stroke decomposition

Edit
Interactive editing using our stroke decomposition

Input

Edit
Pipeline

Input

Preprocess

Extract Layers

Edit
Pipeline

Input
Preprocess
Extract Layers
Edit
Preprocessing Overview
Preprocessing Overview
Preprocessing Overview
Preprocessing Overview
The value of an unblocked pixel should be piecewise constant in time (stable)
The value of an unblocked pixel should be piecewise constant in time (stable).

Identical sequences of stable frames provide checkpoints for the painting progress.
Preprocessing
Preprocessing
Preprocessing
Preprocessing
Preprocessing

• See paper for:
  • illumination
  • color shift
  • noise removal
    • 1D $L_0$ smoothing and bilateral filtering
Preprocessing

• See paper for:
  • illumination
  • color shift
  • noise removal
    • 1D $L_0$ smoothing
      and
      bilateral filtering
Recovering Layers

before + ? = after
Recovering Layers

before + ? = after

opaque solution  our solution
Recovering Layers
Recovering Layers

|----------------|--------------------|---------------------|
## Recovering Layers

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The standard for:</td>
<td>digital compositing</td>
<td>physical compositing</td>
</tr>
</tbody>
</table>
# Recovering Layers

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The standard for:</td>
<td>digital compositing</td>
<td>physical compositing</td>
</tr>
<tr>
<td>Compositing operation:</td>
<td>Linear</td>
<td>Non-linear</td>
</tr>
</tbody>
</table>
# Recovering Layers

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The standard for:</td>
<td>digital compositing</td>
<td>physical compositing</td>
</tr>
<tr>
<td>Compositing operation:</td>
<td>Linear</td>
<td>Non-linear</td>
</tr>
<tr>
<td>Used in graphics:</td>
<td>Almost everywhere</td>
<td>Occasionally Lu et al. [2014], ...</td>
</tr>
</tbody>
</table>
Porter-Duff Model

- “Over” operator:

\[
After = Before \cdot (1 - \alpha) + Paint \cdot \alpha
\]
Porter-Duff Model

• “Over” operator:

\[ After = Before \cdot (1 - \alpha) + Paint \cdot \alpha \]
Porter-Duff Model

• “Over” operator:

\[ \text{After} = \text{Before} \cdot (1 - \alpha) + \text{Paint} \cdot \alpha \]
Porter-Duff Model

• “Over” operator:

\[ \text{After} = \text{Before} \cdot (1 - \alpha) + \text{Paint} \cdot \alpha \]
Porter-Duff Model

- "Over" operator:

\[
After = Before \cdot (1 - \alpha) + Paint \cdot \alpha
\]
Porter-Duff Model

before

after
Porter-Duff Model

before

after

RGB Color Space
Porter-Duff Model

before

after

RGB Color Space
Porter-Duff Model
Porter-Duff Model

Find solution that minimizes $\alpha$

RGB Color Space
Porter-Duff Model

Find solution that minimizes \( \alpha \)

RGB Color Space
Porter-Duff Model

before + Layer (RGBA) = after
Kubelka-Munk Model

- Layer model (mixing model can be found in paper)
Kubelka-Munk Model

- Layer model (mixing model can be found in paper)
Kubelka-Munk Model

- Layer model (mixing model can be found in paper)
Kubelka-Munk Model

- Layer model (mixing model can be found in paper)
Kubelka-Munk Model

- Layer model (mixing model can be found in paper)
Kubelka-Munk Model

- Layer model (mixing model can be found in paper)
Kubelka-Munk Model

- Layer model (mixing model can be found in paper)

\[ \text{Reflectance}_{\text{paint}}: \]

\[ \text{Transmittance}_{\text{paint}}: \]
Kubelka-Munk Model

- Layer model (mixing model can be found in paper)

\[
\text{Find solution that maximizes } \text{Transmittance}_\text{paint}
\]
Kubelka-Munk Model

- Layer model (mixing model can be found in paper)

Find solution that maximizes $\text{Transmittance}^{\text{paint}}$
Kubelka-Munk Model

before

after

Reflectance

Transmittance

Layer (on white canvas)
Results Overview
Results Overview
Editing

• Temporal-Spatial Selection:
Editing

- Coloring using Time Gradient:
Editing

Interactive editing using our stroke decomposition
Editing

Interactive editing using our stroke decomposition
Editing
Conclusion

- A preprocessing method to get a clean, albedo video
Conclusion

- A preprocessing method to get a clean, albedo video
Conclusion

- Two types of solutions for extracting translucent layers
Conclusion

- Two types of solutions for extracting translucent layers
Conclusion

- Useful layers for editing
Conclusion

• Useful layers for editing
Future Work
Future Work

- Camera and canvas calibration.
Future Work

- Camera and canvas calibration.
Future Work

- Camera and canvas calibration.
- Single image layer extraction?
Future Work

• Camera and canvas calibration.
• Single image layer extraction?
• Apply layer data into more systems.
  • WetPaint [Bonanni et al. 2009]
  • Chronicle [Grossman et al. 2010]
  • …
Future Work

• Camera and canvas calibration.
• Single image layer extraction?
• Apply layer data into more systems.
  • WetPaint [Bonanni et al. 2009]
  • Chronicle [Grossman et al. 2010]
  • …
• Apply our technique to art education.
Thank You!

• Contact Information
  • Jianchao Tan: jtan8@gmu.edu
  • Marek Dvorožňák: dvoromar@fel.cvut.cz
  • Daniel Sýkora: sykorad@fel.cvut.cz
  • Yotam Gingold: ygingold@gmu.edu

• Project Website: https://cs.gmu.edu/~ygingold/timemap/

• Artists: Marcello Barenghi, Matyáš Veselý, Dani Jones, semisecretsoftware (YouTube)

• Sponsors:
  • United States National Science Foundation, Google.
  • Technology Agency of the Czech Republic, Czech Science Foundation, Grant Agency of the Czech Technical University in Prague
P-D and K-M Comparison

Layers

P-D

K-M

Input
P-D and K-M Comparison

Layers

P-D

K-M

Input
P-D and K-M Comparison

Layers

P-D

K-M

Input
P-D and K-M Comparison

Layers

P-D

K-M

Input
Preprocessing Comparison

0.1
[Godbehere et al. 2012]

0.3
[Zivkovic and van der Heijden 2006]

0.5

0.7

0.9

[Godbehere et al. 2012]

Our method

Repaired frame
Closest-Paint Method