More Optimal Image Processing by Fractional Order Differentiation and Fractional Order Partial Differential Equations

*Dali Chen, Dingyu Xue, YangQuan Chen*

*yqchen@ieee.org, ychen53@ucmerced.edu*

**ME/EECS/SNRI/UCSolar**

**MESA LAB, School of Engineering, University of California, Merced, USA**

*xuedingyu@ise.neu.edu.cn, chendali@ise.neu.edu.cn*

**Information Science and Engineering, Northeastern University, Shenyang 110004, P R China**
Who cares?

- Minimal dose biomedical imaging
- More optimal

Strategies for Reducing Radiation Dose in CT (McCollough 2009)
FC for what?

- Better than the best
- New sciences
- Need killing apps.
UC Merced

- The Research University of the Central Valley
- Central Located
  - Sacramento – 2 hrs
  - San Fran. – 2 hrs
  - Yosemite – 1.5 hrs
  - LA – 4 hrs
- Surrounded by farmlands and sparsely populated areas
UC Merced

- Established 2005
- 1st research university in 21st century in USA.
- 6,000 Undergraduates
- 300 Grads (200+ Ph.D)
- Strong Undergraduate Research Presence (HSI, MSI)
http://mechatronics.ucmerced.edu

- **Mechatronics, Embedded Systems and Automation**
  - Backup name: *Mechatronics, Energy Systems and Autonomy*
  - ASME DED, MESA TC. http://iel.ucdavis.edu/mesa/
    - 2013 MESA conference: Portland, OR
      http://www.asmeconferences.org/IDETC2013/
MESA Labs

- Director: Dr. YangQuan Chen
- 4 Ph.D. Students
- 1 MSc student
- 20+ Undergraduates
- 4 Visiting Ph.D. Students
- 2 Visiting Professors
- Short term visiting students
- 3 3D printers
MESA Research Areas/Strengths

- Unmanned Aerial Systems and UAV-based Personal Remote Sensing (PRS)
- Cyber-Physical Systems (CPS)
- Modeling and Control of Renewable Energy Systems
- Mechatronics
- Applied Fractional Calculus (AFC)
Roadmap: More Optimal Image Processing

\[ u_0 = u + n; \quad u_0 \text{ is the given image; } \quad n \text{ is the noise.} \]

\[ \hat{u} = \arg \max_u p(u|u_0) = \arg \max_u p(u)p(u_0|u) \]

\[ \min E[u|u_0] = \frac{1}{2} \int_\Omega |\nabla u|^2 dx + \frac{\lambda}{2} \int_\Omega (u - u_0)^2 dx \]

\[ -\Delta u + \lambda u = \lambda u_0 \quad \text{Euler-Lagrange Equation} \]

\[ u_0 = u + \frac{-\Delta u}{\lambda} = u + \omega \quad \text{Detail information} \]
**Introduction: Optimal Image Processing**

\[ u_0 = u + n; \quad u_0 \text{ is the given image;} \quad n \text{ is the noise.} \]

\[ \hat{u} = \arg \max_u p(u|u_0) = \arg \max_u p(u)p(u_0|u) \]

\[ \min E[u|u_0] = \frac{1}{2} \int_{\Omega} |\nabla u|^2 \, dx + \frac{\lambda}{2} \int_{\Omega} (u - u_0)^2 \, dx \]

\[ -\Delta u + \lambda u = \lambda u_0 \quad \text{Euler-Lagrange Equation} \]

\[ u_0 = u + \frac{-\Delta u}{\lambda} = u + \omega \quad \text{Detail information} \]
Outline

- Fractional Order Image Enhancement
- Fractional Order Image Edge Detection
- Fractional Order Image Denoising
- Fractional Order Image Segmentation
- Fractional Order Optical Flow
Fractional Order Image Enhancement

- **Aim of Image Enhancement**\(^{[1]}\):
  - Enhance the contrast and detail information
  - Easy for observation
  - Easy for subsequent processing
Problem Description

$\alpha = 0$

$\alpha = 1$

$\alpha = 0.5$
Digital Fractional Order Savitzky-Golay Differentiator \cite{2}

\[
\hat{Y}_i^{(\alpha)} = X_i^{(\alpha)} B = W_i^{(\alpha)} Y \\
= \left[ \frac{1}{\Gamma(1 - \alpha)} i^{-\alpha}, \frac{1}{\Gamma(2 - \alpha)} i^{1-\alpha}, \frac{\Gamma(3)}{\Gamma(3 - \alpha)} i^{2-\alpha}, \ldots, \frac{\Gamma(n + 1)}{\Gamma(n + 1 - \alpha)} i^{n-\alpha} \right] (X^T X)^{-1} X^T Y,
\]  

(1)

- \( Y \): input signal;
- \( I \): filtering window size;
- \( n \): degree of polynomial function;
- \( i = 1, 2, \ldots, I \).

Good at dealing with noisy signal
Extend to 2-Dimension$^{[3]}$

Assume $l=2m+1$ and

$$W_{m+1}^{(\alpha)} = [W(1), W(2), \ldots, W(2m + 1)]$$  \hspace{1cm} (2)

Then, 2-D DFOSGD templates:

\[ W_{x+}^{(\alpha)} \]

\[
\begin{array}{cccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & \cdots & 0 \\
w(1) & \cdots & w(m) & \cdots & w(2m-1) \\
0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

\[ W_{x-}^{(\alpha)} \]

\[
\begin{array}{cccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & \cdots & 0 \\
w(1) & \cdots & w(m) & \cdots & w(2m-1) \\
0 & \cdots & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

\[ W_{y+}^{(\alpha)} \]

\[
\begin{array}{cccccc}
0 & 0 & w(1) & 0 & 0 & 0 \\
0 & \cdots & 0 \\
w(1) & \cdots & w(m) & \cdots & w(1) \\
0 & \cdots & 0 \\
0 & 0 & w(2m-1) & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

\[ W_{y-}^{(\alpha)} \]

\[
\begin{array}{cccccc}
0 & 0 & w(2m-1) & 0 & 0 & 0 \\
0 & \cdots & 0 \\
w(1) & \cdots & w(m) & \cdots & w(1) \\
0 & \cdots & 0 \\
0 & 0 & w(2m-1) & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
\end{array}
\]
Extend to 2-Dimension
Calculate the $\alpha$th order derivatives of $G(x, y)$ in the different directions by Eq. (3).

$G_{x+}^{(\alpha)} (x, y) = \sum_{k=-m}^{m} \sum_{l=-m}^{m} W_{x+}^{(\alpha)} (k, l) G(x - k, y - l),$

$G_{x-}^{(\alpha)} (x, y) = \sum_{k=-m}^{m} \sum_{l=-m}^{m} W_{x-}^{(\alpha)} (k, l) G(x - k, y - l),$

$G_{y+}^{(\alpha)} (x, y) = \sum_{k=-m}^{m} \sum_{l=-m}^{m} W_{y+}^{(\alpha)} (k, l) G(x - k, y - l),$

$G_{y-}^{(\alpha)} (x, y) = \sum_{k=-m}^{m} \sum_{l=-m}^{m} W_{y-}^{(\alpha)} (k, l) G(x - k, y - l),$

$G_{\downarrow}^{(\alpha)} (x, y) = \sum_{k=-m}^{m} \sum_{l=-m}^{m} W_{\downarrow}^{(\alpha)} (k, l) G(x - k, y - l),$

$G_{\uparrow}^{(\alpha)} (x, y) = \sum_{k=-m}^{m} \sum_{l=-m}^{m} W_{\uparrow}^{(\alpha)} (k, l) G(x - k, y - l),$

$G_{\downarrow}^{(\alpha)} (x, y) = \sum_{k=-m}^{m} \sum_{l=-m}^{m} W_{\downarrow}^{(\alpha)} (k, l) G(x - k, y - l),$

G: input image;
Implementation

Image enhancing algorithm flow

- **Step 1:** Calculate:

\[ G^{(\alpha)}_d(x, y), \ d \in \Omega, \ \Omega := \{x^+, y^+, x^-, y^-, \downarrow, \uparrow, \rightarrow, \leftarrow\} \]

- **Step 2:** Calculate:

\[ G^{(\alpha)}(x, y) = Sat(\max\{G^{(\alpha)}_d(x, y)|d \in \Omega\}). \]

Enhanced image

\[
Sat(u) = \begin{cases} 
0, & u < 0 \\
\min(u, L), & u \in [0, L] \\
L, & u > L 
\end{cases}
\]
Lake Tahoe

http://earthobservatory.nasa.gov/

Experiments

Snow-covered volcanoes on Russia Kamchatka Peninsula
Experiments

Moon

http://bf-astro.com/

Orion Nebula
How to Choose Fractional Order?

\[ \alpha = 0.1 \]

\[ \alpha = 0.4 \]

\[ \alpha = 0.8 \]
Unsupervised optimization algorithm

Let: $\Omega^{(\alpha)} := \{(x, y) | G^{(\alpha)}(x, y) = L\}$

$N^{(\alpha)}$ is the size of $\Omega^{(\alpha)}$

- $N^{(\alpha)}$ increases when $\alpha$ increases.
- Image is under-enhanced when $N^{(\alpha)}$ is small.
- Image is over-enhanced when $N^{(\alpha)}$ is large.

$\hat{\alpha} = \arg\min_{\alpha} J(\alpha)$  \hspace{1cm} (4)

$M(x) = \begin{cases} 
\left(\frac{x}{5}\right)^2, & x \in [0, 25) \\
50 - \left(\frac{x - 50}{5}\right)^2, & x \in [25, 150) \\
50, & x \in [50, 150) \\
\frac{250 - x}{2}, & x \in [150, 250) \\
0, & x \in [250, 255) 
\end{cases}$

$J(\alpha) = \sum_{(x, y) \in \Omega^{(\alpha)}} |G^{(\alpha)}(x, y) - G(x, y) - M(avg)| / N^{(\alpha)}$
Experiments

\[ \alpha = 0.32 \]

\[ \alpha = 0.28 \]
Conclusion

- The digital fractional order Savitzky-Golay differentiator is proposed, see [2];
- The fractional order image enhancing method is proposed, see [3];
- An unsupervised optimization algorithm is proposed for choosing the fractional order, see [3];
Fractional Order Image Edge Detection

- First-Order Edge Detector: Roberts, Prewitt and Sobel\textsuperscript{[1]}
  - high false reject rate (FRR)
- Second-Order Edge Detector: Laplacian of Gaussian\textsuperscript{[1]}
  - high false accept rate (FAR)
- Fractional-Order Edge Detector\textsuperscript{[4]}
  - high density noise
## Motivation

- Fractional differential-based approach
- Robust image edge detection
- Accurate
- Immunity to noise

## Outlines

- Implementation
- Analysis
  - Frequency-domain analysis
  - Parameter analysis
- Experiments
  - Evaluation method
  - Comparison analysis
  - Robustness analysis
Implementation

*Fractional differential mask for edge detection*

- Riemann-Liouville fractional integral\(^5\):

\[
a I_t^\alpha f(t) = \frac{1}{\Gamma(\alpha)} \int_a^t (t - \tau)^{\alpha - 1} f(\tau) d\tau, \quad t \in [a, b]
\]

- Rewrite (1) by convolution formula,

\[
a I_t^{1-\alpha} f(t) = \frac{t^{-\alpha} \ast f(t)}{\Gamma(1 - \alpha)} = h(t, \alpha) \ast f(t) \tag{5}
\]
Implementation

**Fractional differential mask for edge detection**

- **Riemann-Liouville fractional derivative:**

\[
\alpha D_{\alpha}^t f(t) = \frac{d}{dt} \alpha I_{\alpha}^{1-\alpha} f(t) = \frac{d}{dt} \left( h(t, \alpha) \ast f(t) \right) = h'(t, \alpha) \ast f(t). \quad (6)
\]

- **Expand to 2-D:**

\[
h(x, y, \alpha) = \frac{(x^2 + y^2)^{-\alpha/2}}{\Gamma(1 - \alpha)} \quad (9)
\]
Implementation

*Fractional differential mask for edge detection*

- 2-D formulas:

\[
\alpha D_x^\alpha f(x, y) = \frac{\partial h(x, y, \alpha)}{\partial x} \ast f(x, y)
\]

\[
= H_x(x, y, \alpha) \ast f(x, y),
\]

(10)

\[
\alpha D_y^\alpha f(x, y) = \frac{\partial h(x, y, \alpha)}{\partial y} \ast f(x, y)
\]

\[
= H_y(x, y, \alpha) \ast f(x, y).
\]

(11)
Implementation

Fractional differential mask for edge detection

FDM equations:

\[ H_x(x, y, \alpha) = -\frac{\alpha}{\Gamma(1-\alpha)}x \left( x^2 + y^2 \right)^{-\alpha/2-1}, \]  \hspace{1cm} (12)

\[ H_y(x, y, \alpha) = -\frac{\alpha}{\Gamma(1-\alpha)}y \left( x^2 + y^2 \right)^{-\alpha/2-1}. \]  \hspace{1cm} (13)
Implementation

Fractional differential mask for edge detection

Discrete FDM equations:

\[ H_x(x_M, y_N, \alpha) = -\frac{\alpha x_M}{\Gamma(1-\alpha)} \left( x_M^2 + y_N^2 \right)^{-\alpha/2-1} \]  \hspace{1cm} (14)

\[ H_y(x_M, y_N, \alpha) = -\frac{\alpha y_N}{\Gamma(1-\alpha)} \left( x_M^2 + y_N^2 \right)^{-\alpha/2-1} \]  \hspace{1cm} (15)

Here \( x_M = -M, -M + 1, \ldots M \) \quad \( y_N = -N, -N + 1, \ldots N \)

\[ H_x(0, 0, \alpha) = 0 \] , and \[ H_y(0, 0, \alpha) = 0 \]
Implementation

Fractional differential mask for edge detection

When $\alpha = 0.5$, $5 \times 5$ FDM:

$$
\begin{pmatrix}
0.0419 & 0.0377 & 0 & -0.0377 & -0.0419 \\
0.0755 & 0.1186 & 0 & -0.1186 & -0.0755 \\
0.0997 & 0.2821 & 0 & -0.2821 & -0.0997 \\
0.0755 & 0.1186 & 0 & -0.1186 & -0.0755 \\
0.0419 & 0.0377 & 0 & -0.0377 & -0.0419 \\
\end{pmatrix}
$$

$$
\begin{pmatrix}
0.0419 & 0.0755 & 0.0997 & 0.0755 & 0.0419 \\
0.0377 & 0.1186 & 0.2821 & 0.1186 & 0.0377 \\
0 & 0 & 0 & 0 & 0 \\
-0.0377 & -0.1186 & -0.2821 & -0.1186 & -0.0377 \\
-0.0419 & -0.0755 & -0.0997 & -0.0755 & -0.0419 \\
\end{pmatrix}
.$$
Implementation

Image edge detection algorithm flow

- **Step 1**: Calculate:
  
  \[ M(x, y) = \sqrt{(D_x^\alpha f(x, y))^2 + (D_y^\alpha f(x, y))^2} \]
  
  \[ \varphi(x, y) = \tan^{-1}(D_y^\alpha f(x, y)/D_x^\alpha f(x, y)) \]

- **Step 2**: Find an edge direction.
- **Step 3**: Non-maximum suppression.
- **Step 4**: Hysteresis Thresholding.
- **Step 5**: Link edge.
function eout=ch_fedge(a,alpha,width)
if ~isa(a, 'double') && ~isa(a, 'single'), a = im2single(a); end
[m,n] = size(a); e = false(m,n); PercentOfPixelsNotEdges = .7;
ThresholdRatio = .4; chw = -alpha/gamma(1-alpha);
[x,y]=meshgrid(-width:width,-width:width);
dgau2D=x.*(x.*x+y.*y).^(-alpha/2-1)*chw;
dgau2D(width+1,width+1)=0;
ax = imfilter(a, dgau2D, 'conv','replicate');
ay = imfilter(a, dgau2D, 'conv','replicate');
mag = sqrt((ax.*ax) + (ay.*ay));magmax = max(mag(:));
if magmax>0,mag = mag / magmax;end  counts=imhist(mag, 64);
highThresh = find(cumsum(counts) > ...  PercentOfPixelsNotEdges*m*n,1,'first') / 64;
lowThresh = ThresholdRatio*highThresh;
thresh = [lowThresh highThresh]; idxStrong = [];
Implementation

Image edge detection algorithm flow

```
for dir = 1:4
    idxLocalMax = cannyFindLocalMaxima(dir,ax,ay,mag);
    idxWeak = idxLocalMax(mag(idxLocalMax) > lowThresh);
    e(idxWeak)=1;
    idxStrong = [idxStrong;...
                 idxWeak(mag(idxWeak) > highThresh)];
end
if isempty(idxStrong)
    rstrong = rem(idxStrong-1, m)+1;
    cstrong = floor((idxStrong-1)/m)+1;
    e = bwmselect(e, cstrong, rstrong, 8);
    e = bwmorph(e, 'thin', 1);
end
eout = e;
```
Analysis

Parameter analysis

- $M = 5$
- $\alpha : 0 \rightarrow 1$

Original image

- Smoothing
- Enhancing
Analysis

Parameter analysis

- $\alpha = 0.9$
- $M : 2 \rightarrow 40$
- Smoothing $\uparrow$
- Enhancing $\downarrow$
Analysis

Parameter and noise immunity

- $M = 4$
- $\alpha : 1 \rightarrow 0$

Noisy image: Gaussian noise with zero-mean and variance $\sigma^2 = 0.01$

- Noise immunity
- Enhancing
Analysis

Parameter and noise immunity

- $\alpha = 0.5$
- Noise immunity
- Enhancing

$M = 2$  $M = 3$  $M = 4$
Analysis

Conclusion

Frequency-domain analysis

Parameter analysis

- $\alpha$
- Smoothing
- Enhancing

$M$
- Smoothing
- Enhancing
Experiments

Evaluation method

- **False Reject Rate (FRR):**
  \[ \mu_{FRR} = \frac{\Psi(A - A \cap B)}{\Psi(A)} \]

- **False Accept Rate (FAR):**
  \[ \mu_{FAR} = \frac{\Psi(\overline{A} \cap B)}{\Psi(B)} \]

- **Single-Pixel-Detecting (SPD):**
  \[ \mu_{SPD} = \frac{\Psi(\varphi(B))}{\Psi(B)} \]

- \( \Psi(X) \) denotes the number of elements in \( X \)
- \( \mu_{FRR} \rightarrow \mu_{FAR} \)
- \( \varphi \) is the typical thinning operator [6]
- Single edge: \( \mu_{SPD} = 1 \)
Experiments: *Comparison analysis*

(a) Multi-scale linear edge image
(b) Robert edge detector
(c) Prewitt edge detector
(d) Sobel edge detector
(e) LoG edge detector
(f) Canny edge detector
(g) Oustaloup edge detector
(h) Proposed edge detector

**Conclusion:**
1. Better than the Roberts, Prewitt, Sobel, LoG and Canny edge detectors;
2. Similar with the Oustaloup edge detector.
Experiments: Comparison analysis

(a) Nonlinear edge image
(b) Robert edge detector
(c) Prewitt edge detector
(d) Sobel edge detector
(e) LoG edge detector
(f) Canny edge detector
(g) Oustaloup edge detector
(h) Proposed edge detector

Conclusion: 1. Better than the Roberts, Prewitt, Sobel and LoG edge detectors; 2. Similar with the Canny and Oustaloup edge detector.
Experiments:  *Comparison analysis*

Table 1: Quantitative comparison among the proposed method and six typical methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Linear</th>
<th>Nonlinear</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\mu_{FR}$</td>
<td>$\mu_{SPD}$</td>
</tr>
<tr>
<td>Roberts</td>
<td>0.0852(4)</td>
<td>0.9844(6)</td>
</tr>
<tr>
<td>Prewitt</td>
<td>0.0887(6)</td>
<td>0.9892(3)</td>
</tr>
<tr>
<td>Sobel</td>
<td>0.0882(5)</td>
<td>0.9887(4)</td>
</tr>
<tr>
<td>LoG</td>
<td>0.1348(7)</td>
<td>0.9850(5)</td>
</tr>
<tr>
<td>Canny</td>
<td>0.0226(3)</td>
<td>0.9949(2)</td>
</tr>
<tr>
<td>Oustaloup</td>
<td>0.0173(2)</td>
<td>0.9991(1)</td>
</tr>
<tr>
<td>Proposed</td>
<td><strong>0.0135(1)</strong></td>
<td>0.9952(2)</td>
</tr>
</tbody>
</table>

$\mu_{FR} = \mu_{FRR} + \mu_{FAR}$
Experiments

Robustness analysis

- **X Axis**
  - Noisy image
  - Signal To Noise Rate (SNR)

- **Y Axis**
  - False rate (FR)

- Red curve
- Better than the other detectors

Fig. FR comparison among the seven edge detectors for the multi-scale linear edge noisy images with different SNRs
Experiments

Robustness analysis

- **X Axis**
  - Noisy image
  - Signal To Noise Rate (SNR)

- **Y Axis**
  - False rate (FR)

- Robustness is better.

Fig. FR comparison among the seven edge detectors for the nonlinear edge noisy images with different SNRs.
Conclusion

- A new fractional differential-based method is proposed for robust image edge detection, see [6];
- Frequency domain analysis;
- Good edge-detecting capability and robustness;
- Fast, real-time systems.
Fractional Order Image Denoising

Problem Description

\[
\min \int_{\Omega} |Du| + (f - u)^2 \, d\Omega \quad \Rightarrow \quad ? \quad \Leftarrow \quad \min \int_{\Omega} |D^2 u| + (f - u)^2 \, d\Omega
\]

\[
\min \int_{\Omega} f(Du) \, d\Omega \quad \Rightarrow \quad \min \int_{\Omega} f(D^\alpha u) \, d\Omega \quad \Leftarrow \quad \min \int_{\Omega} f(D^2 u) \, d\Omega
\]


Blocky effect \quad Uplifting effect

1 \quad 2
Outline

- Majorization-Minimization (MM) Method
- Fractional Order Total Variation (TV)-L2 Model
- Majorization of $TV^\alpha_p(u)$
- Numerical Scheme
- Experiments
Why MM Method?

- **Continuous Domain**\(^{[13-16]}\):
  - Newton’s Method
  - Duality Theory
  - Euler-Lagrange Equation
  - FTVd Method

- **Discrete Domain**\(^{[17-18]}\):
  - Majorization-Minimization (MM) Method
Majorization-Minimization (MM) Method

\[ \min_{x \in \mathbb{R}} E(x) \]

Satisfy:

\[ x_{k+1} = \text{arg min}_{x \in \mathbb{R}} H_k(x) \]

\[ H_k(x) \geq E(x), \quad \forall x \]

\[ H_k(x_k) = E(x_k) \]
Fractional Order TV-L2 Model

\[ \hat{u} = \arg \min_u \left\{ E(u) = \| f - u \|_2^2 + \lambda \text{TV}_p^\alpha (u) \right\} \]

Data term \quad Regularization term

\[
\text{TV}_p^\alpha (u) = \sum_{n \in \Omega} \| D^\alpha u(n) \|_p
\]

\[
\| D^\alpha u(n) \|_1 = |D^\alpha_h u(n)| + |D^\alpha_v u(n)|
\]

\[
\| D^\alpha u(n) \|_2 = \sqrt{(D^\alpha_h u(n))^2 + (D^\alpha_v u(n))^2}
\]

\( \lambda \): regularization parameter;

\( D^\alpha_h \): horizontal fractional order derivative;

\( D^\alpha_v \): vertical fractional order derivative;

\( f \): Noisy image;

\( u \): Clean image;

\( \| u \|_\nu \): \( \nu \)-norm;
Majorization of $\text{TV}_p^\alpha(u)$

Assume

$$Q_{(k,1)}(u) = \frac{1}{2} \text{TV}_1^\alpha(u_k) + \frac{1}{2} u^T (D^\alpha)^T V_{(k,1)}^{-1} D^\alpha u = C_1^\alpha(u_k) + \frac{1}{2} u^T (D^\alpha)^T V_{(k,1)}^{-1} D^\alpha u$$

$D^\alpha = [(D_h^\alpha)^T, (D_v^\alpha)^T]^T$

$V_{(k,1)} = \text{diag}(\Lambda_{h_k}^h, \Lambda_{v_k}^v)$

$\Lambda_{v_k}^v = \text{diag}(|D_v^\alpha u_k|)$

$\Lambda_{h_k}^h = \text{diag}(|D_h^\alpha u_k|)$

$Q_{(k,1)}(u) \geq \text{TV}_1^\alpha(u)$

$Q_{(k,1)}(u_k) = \text{TV}_1^\alpha(u_k)$

$Q_{(k,1)}(u)$ is the majorization of $\text{TV}_1^\alpha(u)$
Majorization of $TV^\alpha_p(u)$

Assume

$$Q_{(k,2)}(u) = TV_2^\alpha(u_k) + \frac{1}{2} u^T (D^\alpha)^T V_{(k,2)}^{-1} D^\alpha u = C_2^\alpha(u_k) + \frac{1}{2} u^T (D^\alpha)^T V_{(k,2)}^{-1} D^\alpha u,$$

$$V_{(k,2)} = \text{diag}(v_k, v_k)$$

$$v_k = \text{diag}\left(\sqrt{(D_h^\alpha u_k)^2 + (D_v^\alpha u_k)^2}\right)$$

$$Q_{(k,2)}(u_k) = TV_2^\alpha(u_k)$$

$$Q_{(k,2)}(u) \geq TV_2^\alpha(u)$$

$Q_{(k,2)}(u)$ is the majorization of $TV_2^\alpha(u)$
Majorization of $\text{TV}_p^\alpha(u)$

The majorization of $\text{TV}_p^\alpha(u)$ is

$$Q_{(k,p)}(u) = C_p^\alpha(u_k) + \frac{1}{2}u^T(D^\alpha)^T V_{(k,p)}^{-1} D^\alpha u.$$  

So, the majorization of the fractional order TV model is

$$H_{(k,p)}(u) = \|f - u\|_2^2 + \frac{1}{2}u^T(D^\alpha)^T V_{(k,p)}^{-1} D^\alpha u + C_p^\alpha(u_k),$$  

constant

Thus, the $u$ can be estimated by solving a sequence of optimization problems

$$u_{k+1} = \arg \min_u \left\{\|f - u\|_2^2 + \frac{\lambda}{2}u^T(D^\alpha)^T V_{(k,p)}^{-1} D^\alpha u\right\}.$$  

It leads to a linear system:

$$(I + \lambda(D^\alpha)^T V_{(k,p)}^{-1} D^\alpha)u_{k+1} = f.$$
Numerical Scheme

1. Let the input image be $f$ and set $k = 0$ and $u_k = f$. Initialize $K_{\sigma'}$, $\sigma$, $M$, $N$ and iteration number $K_{iter}$.

2. Set $u_k = K_{\sigma'} * u_k$ and $\lambda_k^p = MN\sigma^2 / 2TV_p^\alpha(u_k)$.

3. Compute $A_k = (I + \lambda_k^p(D^\alpha)^T V_{(k,p)}^{-1} D^\alpha)$.

4. Compute $u_{k+1}$ by using the conjugate gradient algorithm to solve the linear system $A_k u_{k+1} = f$.

5. If $k = K_{iter}$, stop; else, set $k = k + 1$ and go to step 2.
Experiments: Restraint of block effect

Conclusion: the proposed fractional order TV-L2 model can reduce blocky effect.
Experiments: Analysis of denoising performance

PSNR: peak signal to noise ratio

<table>
<thead>
<tr>
<th>Image</th>
<th>SD</th>
<th>IP-M</th>
<th>F-O-PDE</th>
<th>IF-O-PDE</th>
<th>ROF</th>
<th>$TV_2^\alpha$-L^2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Barbara</td>
<td>10</td>
<td>31.2427</td>
<td>29.3776</td>
<td>29.3777</td>
<td>31.0871</td>
<td><strong>31.3792</strong></td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>24.4496</td>
<td>22.9267</td>
<td>22.9273</td>
<td>24.7415</td>
<td><strong>24.8166</strong></td>
</tr>
<tr>
<td>Lena</td>
<td>10</td>
<td>33.6393</td>
<td>31.5440</td>
<td>31.5442</td>
<td>33.8378</td>
<td><strong>34.7437</strong></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>29.7665</td>
<td>27.8080</td>
<td>27.8098</td>
<td>30.4101</td>
<td><strong>31.4387</strong></td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>27.4437</td>
<td>26.0883</td>
<td>26.1019</td>
<td>28.5889</td>
<td><strong>29.6095</strong></td>
</tr>
<tr>
<td>Peppers</td>
<td>10</td>
<td>33.6967</td>
<td>31.8199</td>
<td>31.8205</td>
<td>33.8715</td>
<td><strong>33.9328</strong></td>
</tr>
</tbody>
</table>

IP-M: Improved Perona and Malik model [7]; F-O-PDE: fourth order PDE model [8]; IF-O-PDE: improved fourth order PDE model[19]; ROF model [9]; proposed fractional order TV2-L2 model.

Conclusion: the denoising performance proposed fractional order TV$_2$-L2 model is better than the other four methods.
Experiments: Analysis of denoising performance

Table 1: PSNR quantitative comparison between $TV_1^\alpha-L^2$ and $TV_2^\alpha-L^2$ model

<table>
<thead>
<tr>
<th>Model</th>
<th>Barbara 10</th>
<th>Barbara 20</th>
<th>Barbara 30</th>
<th>Lena 10</th>
<th>Lena 20</th>
<th>Lena 30</th>
<th>Peppers 10</th>
<th>Peppers 20</th>
<th>Peppers 30</th>
<th>Goldhill 10</th>
<th>Goldhill 20</th>
<th>Goldhill 30</th>
</tr>
</thead>
</table>

Conclusion: the denoising performance proposed fractional order $TV_2-L^2$ model is better than fractional order $TV_1-L^2$ model.
Experiments: Analysis of denoising performance

Relation between the iteration number and PSNR value

Conclusion: the method is stable after 15 iterations.
Experiments:

Lung nodule segmentation experiment

Noisy lung nodule CT image

Partial enlarged view of the segmented result

TV$_2^1 - L^2$

TV$_2^{1.5} - L^2$

Conclusion: 1. fractional order TV-L2 denoising method is helpful for improving the accuracy of the post-processing technologies in the lung nodule CT image segmentation.
2. the detected edge of TV$_2^{1.5} - L^2$ is more accurate than that of TV$_2^1 - L^2$
Experiments:

Cardiac muscular segmentation experiment

Conclusion: 1. fractional order TV-L2 denoising method is helpful for improving the accuracy of the post-processing technologies in the cardiac muscular PET image segmentation. 2. the detected edge of $TV^{1.5}_2 - L^2$ is more accurate than that of $TV^{1}_2 - L^2$.
Two fractional order TV-L2 models are constructed, see [12];

Majorization-minimization algorithm was used to solve fractional TV optimization problem, see [12];

Majorizers of two fractional order TV regularizers are obtained in one uniform formula, see [12];

Avoid the blocky effect.

Conclusion
Fractional Order Level Set Model

Introduction

What is the Level Set Method?

LSM is a numerical technique for tracking interfaces and shapes.

What is Level Set Good for? [20-27]

- Image Segmentation
- Tracking
- Computer Graphics
- Computational Geometry
- Computational fluid dynamics
Fractional Order Level Set Model

Problem Description

\[ \min \int_\Omega \left( \lambda_1 |I - c_1|^2 H(\phi) + \lambda_2 |I - c_2|^2 (1 - H(\phi)) + \mu \delta(\phi) |D^\alpha \phi| \right) d\Omega \]

C-V Model \[20\] 2001

\[ \min E(f_1(x), f_2(x), \phi) = \lambda_1 \int_\Omega \left( \int_\Omega K_{\sigma}(x \rightarrow y) I(y) - f_1(x) H(\phi(y)) dy \right) dx \]
\[ + \lambda_2 \int_\Omega \left( \int_\Omega K_{\sigma}(x \rightarrow y) I(y) - f_1(x) \right)^2 (1 - H(\phi(y))) dy dx \]
\[ + \mu \int_\Omega \delta(\phi) |D^\alpha \phi| d\Omega \]

RSF Model \[21\] 2008
Outline

- C-V Model
- Fractional Order C-V Model
- RSF (Region-Scalable Fitting) Model
- Fractional Order RSF Model
- Numerical Algorithm
- Experiments
C-V Model \textsuperscript{[20]}

Model description:

\[
\min \int_{\Omega} (\lambda_1 |I - c_1|^2 H(\phi) + \lambda_2 |I - c_2|^2 (1 - H(\phi)) + \mu \delta(\phi)|D\phi|) d\Omega
\]

\[
c_1(\phi) = \frac{\int_{\Omega} I(x, y) H(\phi(x, y)) dx dy}{\int_{\Omega} H(\phi(x, y)) dx dy} \quad (17)
\]

\[
H(\phi) = \begin{cases} 
1, & \phi \geq 0 \\
0, & \phi < 0 
\end{cases}
\]

\[
c_2(\phi) = \frac{\int_{\Omega} I(x, y) (1 - H(\phi(x, y))) dx dy}{\int_{\Omega} (1 - H(\phi(x, y))) dx dy} \quad (18)
\]

\(
\lambda_1 \text{ and } \lambda_2 \text{ are positive constants}
\)
C-V Model

\[ \phi(x, y, t) = \begin{cases} 
D(x, y, t), & (x, y) \in \Omega_1 \\
0, & (x, y) \in C \\
-D(x, y, t), & (x, y) \in \Omega_2 
\end{cases} \]

\(D(x, y, t)\) is the Euclidean distance from point \((x, y)\) to curve \(C\) at time \(t\).

\(\Omega_1\) is inside \(C\)

\(\Omega_2\) is outside \(C\)

\(\phi\) is the level set.
Fractional Order C-V Model

Model description:

\[
\min \int_{\Omega} (\lambda_1 |I - c_1|^2 H(\phi) + \lambda_2 |I - c_2|^2 (1 - H(\phi)) + \mu \delta(\phi) |D^\alpha \phi|) d\Omega
\]

Euler–Lagrange Equation:

\[
\frac{\partial \phi}{\partial t} = \delta(\phi) [\mu \text{div}^\alpha \left( \frac{D^\alpha \phi}{|D^\alpha \phi|} \right) - \lambda_1 (I - c_1)^2 + \lambda_2 (I - c_2)^2]
\]
RSF Model $^{[21]}$

Model description: Region-Scalable Fitting Energy

$$
\min E(f_1(x), f_2(x), \phi) = \lambda_1 \int \left( \int_{\Omega} K_\sigma(x - y) |I(y) - f_1(x)|^2 H(\phi(y)) dy \right) dx
$$

$$
+ \lambda_2 \int \left( \int_{\Omega} K_\sigma(x - y) |I(y) - f_1(x)|^2 (1 - H(\phi(y))) dy \right) dx
$$

$$
+ \mu \int_{\Omega} \delta(\phi) |D\phi| d\Omega \quad \text{Curve Length}
$$

$$
K_\sigma(x - y) = \frac{1}{(2\pi)^{n/2} \sigma^n} \exp\left(-\frac{|x - y|^2}{2\sigma^2}\right)
$$

$I(y)$ denotes the intensity of the point $y$ in a neighborhood of $x$

$f_1$ and $f_2$ are two values that approximate image intensities in $\Omega_1$ and $\Omega_2$
Fractional Order RSF Model

Model description:

\[
\min E(f_1(x), f_2(x), \phi) = \lambda_1 \int \left( \int_{\Omega} K_\sigma(x - y)|I(y) - f_1(x)|^2 H(\phi(y)) dy \right) dx \\
+ \lambda_2 \int \left( \int_{\Omega} K_\sigma(x - y)|I(y) - f_1(x)|^2 (1 - H(\phi(y))) dy \right) dx \\
+ \mu \int_{\Omega} \delta(\phi)|D^\alpha \phi| d\Omega
\]

\[
\frac{\partial \phi}{\partial t} = \delta(\phi)[\mu \text{div}^\alpha \left( \frac{D^{\alpha \phi}}{|D^{\alpha \phi}|} \right) - \lambda_1 e_1 + \lambda_2 e_2] \quad (20)
\]

\[
e_i(x) = \int K_\sigma(y - x)|I(x) - f_i(y)|^2 dy, \quad i = 1, 2
\]
Principal step of algorithm

The principal steps of the algorithm are:

1. Initialize $\phi^0$ by $\phi_0$, $n = 0$;

2. Fractional-Order C-V Model: Compute $c_1(\phi^n)$ and $c_2(\phi^n)$ by (17) and (18);
   Fractional-Order RSF Model: Compute $f_1(x), f_2(y)$ and $K_\sigma(x - y)$;

3. Solve the Euler-Lagrange Equation in $\phi$ from (19) or (20), to obtain $\phi^{n+1}$;

4. Reinitialize $\phi$ locally to the signed distance function to the curve;

5. Check whether the solution is stationary. If not, $n = n + 1$ and repeat.
Conclusion: For intensity homogeneity PET cardiac muscular image, the detected edge of fractional order CV model is more accurate than that of typical CV model.
Conclusion: For intensity homogeneity PET cardiac muscular image, the detected edge of fractional order RSF model is more accurate than that of both fractional order and typical CV models.
Intensity Homogeneity PET Cardiac Muscular Segmentation Demo

Fractional Order Level Set Model

Demo: Intensity homogeneity PET cardiac muscular image

YangQuan Chen, Dingyu Xue, Dali Chen, Congrong Zhang

yangquan.chen@ucmerced.edu
yangchen@iece.org

MESA(Mechatronics, Embedded Systems and Automation)LAB
School of Engineering, University of California, Merced, 5200 N Lake Road, Merced, CA 95343, USA

xuedingyu@ise.neu.edu.cn,
chendali@ise.neu.edu.cn

Information Science and Engineering
Northeastern University
Shenyang 110004, P R China
Conclusion: For intensity inhomogeneity PET cardiac muscular image, both fractional order and typical CV models are not able to get satisfactory result.
Conclusion: For intensity inhomogeneity PET cardiac muscular image, the detected edge of fractional order RSF model is more accurate than that of both fractional order and typical CV models.
Intensity Inhomogeneity PET Cardiac Muscular Segmentation Demo

Fractional Order Level Set Model

Demo: Intensity inhomogeneity PET cardiac muscular image

YangQuan Chen, Dingyu Xue, Dali Chen, Congrong Zhang

yangquan.chen@ucmerced.edu
yqchen@ieee.org
MESA(Mechatronics, Embedded Systems and Automation)LAB
School of Engineering, University of California, Merced, 5200 N Lake Road, Merced, CA 95343, USA

xuedingyu@ise.neu.edu.cn,
chendali@ise.neu.edu.cn
Information Science and Engineering Northeastern University Shenyang 110004, P R China
Conclusion

- Two fractional-order image segmentation model are proposed;
- The effective numerical algorithms are proposed;
- Improved accuracy;
- Provide a new tool for image segmentation.
Fractional Order Optical Flow

Introduction

What is the Optical Flow Problem?

Input: two or more frames of an image sequence;  
Output: displacement field between two consecutive frames, optical flow.

What is Optical Flow Good for? [28-35]

- Robot navigation
- Tracking
- Action recognition
- Video compression
- Stereo reconstruction
- Medical image registration
Fractional Order Optical Flow

Problem Description

\[ \min_u \int_\Omega ((I_x u_1 + I_y u_2 + I_t)^2 + \lambda \| D^\alpha u \|_2^2) d\Omega \]

\[ \min_u \int_\Omega ((I_x u_1 + I_y u_2 + I_t)^2 + \lambda \| D u \|_2^2) d\Omega \]

\[ \min_u \int_\Omega (|\rho(u)| + \lambda \sum_{d=1}^2 \| D^2 u_d \|) d\Omega \]

H-S Model \textsuperscript{[36]} 1981

Second-Order Model \textsuperscript{[11]} 2008

1

Horn-Schunck

2
Outline

- Fractional-Order Optical Flow Model
- Numerical Algorithm
  - Euler–Lagrange Equation
  - Fractional-Order Differential Operator
  - Discrete Euler-Lagrange Equations
  - Structure of the Linear System
  - Multi-scale Approach
- Experiments
Outline

- Improved Fractional-Order Optical Flow Model
  - Discrete Formulation
  - Saddle-Point Formulation

- Numerical Scheme

- Experiments
**Fractional-Order Variational Optical Flow Model** \(^{[37]}\)

Model description:

\[
\min_w \int_\Omega ((I_x u + I_y v + I_t)^2 + \lambda (|D^\alpha u|^2 + |D^\alpha v|^2)) d\Omega
\]

\(I_0(x, y, t)\) is the given image;

\((x, y)^T\) denotes the location with a rectangular image domain \(\Omega \in \mathbb{R}\);

\(w(x, y) := (u(x, y), v(x, y), 1)^\top\) is the optic flow field;

\(I_x, I_y\) and \(I_t\) are the gradients of image sequence \(I\) in \(x, y\) and \(t\);

\(D^\alpha := (D_x^\alpha, D_y^\alpha)^\top\);

\[|D^\alpha u| = \sqrt{(D_x^\alpha u)^2 + (D_y^\alpha u)^2}.\]
Consider an energy function \( J(u, v) \) defined by

\[
J(u, v) = \int_{\Omega} \left( (I_x u + I_y v + I_t)^2 
+ \lambda (|D_x^\alpha u|^2 + |D_y^\alpha u|^2 + |D_x^\alpha v|^2 + |D_y^\alpha v|^2) \right) d\Omega.
\]

Assume that \( u^*(x, y) \) and \( v^*(x, y) \) are the desired functions. Take any test functions \( \eta(x, y) \in C^\infty(\Omega) \) and \( \zeta(x, y) \in C^\infty(\Omega) \) and \( \epsilon \in \mathbb{R} \). We have

\[
\begin{align*}
\quad u(x, y) &= u^*(x, y) + \epsilon \eta(x, y), \\
\quad v(x, y) &= v^*(x, y) + \epsilon \zeta(x, y).
\end{align*}
\]

\[
\begin{align*}
D_x^\alpha u(x, y) &= D_x^\alpha u^*(x, y) + \epsilon D_x^\alpha \eta(x, y), \\
D_y^\alpha u(x, y) &= D_y^\alpha u^*(x, y) + \epsilon D_y^\alpha \eta(x, y), \\
D_x^\alpha v(x, y) &= D_x^\alpha v^*(x, y) + \epsilon D_x^\alpha \zeta(x, y), \\
D_y^\alpha v(x, y) &= D_y^\alpha v^*(x, y) + \epsilon D_y^\alpha \zeta(x, y).
\end{align*}
\]
Euler–Lagrange Equation

\[ J(\epsilon) = \int_{\Omega} \left( (I_x u^* + \epsilon I_x \eta + I_y v^* + \epsilon I_y \zeta + I_t)^2 + \lambda \left( |D_x^\alpha u^* + \epsilon D_x^\alpha \eta|^2 + |D_y^\alpha u^* + \epsilon D_y^\alpha \eta|^2 \right. \right. \]
\[ \left. \left. + |D_x^\alpha v^* + \epsilon D_x^\alpha \zeta|^2 + |D_y^\alpha v^* + \epsilon D_y^\alpha \zeta|^2 \right) \right) d\Omega. \]

Differentiating (21) with respect to \( \epsilon \), we obtain

\[ J'(0) = \int_{\Omega} \left( \eta (MI_x + \lambda (D_x^* D_x^\alpha u^* + D_y^* D_y^\alpha u^*)) \right. \]
\[ \left. + \zeta (MI_y + \lambda (D_x^* D_x^\alpha v^* + D_y^* D_y^\alpha v^*)) \right) d\Omega. \]

\[(I_x u^* + I_y v^* + I_t)I_x + \lambda (D_x^* D_x^\alpha u^* + D_y^* D_y^\alpha u^*) = 0.\]
\[(I_x u^* + I_y v^* + I_t)I_y + \lambda (D_x^* D_x^\alpha v^* + D_y^* D_y^\alpha v^*) = 0.\]

\( D^\alpha^* \) is the right Riemann-Liouville fractional derivative.
Fractional-Order Differential Operator

Let: \( w_0^{(\alpha)} = 1, \quad w_k^{(\alpha)} = (1 - \frac{\alpha + 1}{k}) w_{k-1}^{(\alpha)}, \quad k = 1, 2, \ldots. \)

We have:
\[
D_x^\alpha D_x^\alpha u(i, j) = \sum_{k=-\infty}^{0} w_k^{(\alpha)} u(i - k, j) + \sum_{k=0}^{\infty} w_k^{(\alpha)} u(i - k, j). \tag{22}
\]

Since \( \sum_{k=0}^{\infty} w_k^{(\alpha)} = 0 \), the equation can be rewritten by
\[
D_x^\alpha D_x^\alpha u(i, j) = \sum_{k=-\infty}^{-1} w_k^{(\alpha)} \nabla u(i - k, j) + \sum_{k=1}^{\infty} w_k^{(\alpha)} \nabla u(i - k, j), \tag{22}
\]

where \( \nabla u(i - k, j) = u(i - k, j) - u(i, j) \).
Fractional-Order Differential Operator

For application, we approximate (22) using the following formula:

\[ D^\alpha_x D_x u(i, j) \approx \sum_{k=-L}^{1} w^{(\alpha)}_{|k|} \nabla u(i - k, j) + \sum_{k=1}^{L} w^{(\alpha)}_{k} \nabla u(i - k, j), \quad (23) \]

Similarly, we can obtain

\[ D^\alpha_y D_y u(i, j) \approx \sum_{k=-L}^{1} w^{(\alpha)}_{|k|} \nabla u(i, j - k) + \sum_{k=1}^{L} w^{(\alpha)}_{k} \nabla u(i, j - k). \quad (24) \]
From (23) and (24), the concise discrete formula of the fractional-order differential operator can be described by

\[ D_x^\alpha D_x^\alpha u(i, j) + D_y^\alpha D_y^\alpha u(i, j) \approx \sum_{(\tilde{i}, \tilde{j}) \in \chi(i, j)} w_{k_{\tilde{i}, \tilde{j}}}^{(\alpha)} (u(\tilde{i}, \tilde{j}) - u(i, j)), \]

\[ D_x^\alpha D_x^\alpha v(i, j) + D_y^\alpha D_y^\alpha v(i, j) \approx \sum_{(\tilde{i}, \tilde{j}) \in \chi(i, j)} w_{k_{\tilde{i}, \tilde{j}}}^{(\alpha)} (v(\tilde{i}, \tilde{j}) - v(i, j)). \]

\( \chi(i, j) \) denotes the set of neighbors of pixel \((i, j)\) in axis \(x\) and \(y\); 
\( k_{\tilde{i}, \tilde{j}} \) can be obtained by \( \max(|\tilde{i} - i|, |\tilde{j} - j|) \).
Discrete Euler-Lagrange Equations

Let:

\[ I_{xx}(i, j) = I_x(i, j) \times I_x(i, j), \quad I_{yy}(i, j) = I_y(i, j) \times I_y(i, j), \]
\[ I_{xy}(i, j) = I_x(i, j) \times I_y(i, j), \quad I_{tx}(i, j) = I_t(i, j) \times I_x(i, j), \]
\[ I_{ty}(i, j) = I_t(i, j) \times I_y(i, j). \]

The discrete Euler-Lagrange equations can finally be written as

\[ I_{xx}(i, j)u(i, j) + I_{xy}(i, j)v(i, j) + \lambda \sum_{(\tilde{i}, \tilde{j}) \in \chi(i, j)} w_{k_{\tilde{i}\tilde{j}}}^{(\alpha)}(u(\tilde{i}, \tilde{j}) - u(i, j)) = -I_{tx}(i, j), \]
\[ I_{xy}(i, j)u(i, j) + I_{yy}(i, j)v(i, j) + \lambda \sum_{(\tilde{i}, \tilde{j}) \in \chi(i, j)} w_{k_{\tilde{i}\tilde{j}}}^{(\alpha)}(v(\tilde{i}, \tilde{j}) - v(i, j)) = -I_{ty}(i, j). \]

(25)
Structure of the Linear System

The linear system can be written as:

\[ AX = -B \]

\[ A = \begin{pmatrix} I_{xx} - \lambda D^{\alpha^*} D^{\alpha} & I_{xy} \\ I_{yx} & I_{yy} - \lambda D^{\alpha^*} D^{\alpha} \end{pmatrix} \]

\[ X = [U, V]^T \quad B = [I_{tx}, I_{ty}]^T \]

It can be solved by many typical methods such as the Jacobi method, the Gauß–Seidel method, the successive overrelaxation method and the preconditioned conjugate gradient (PCG) method.
Multi-scale Approach

Good for finding the global minimum.
The accuracy of optical flow estimation algorithms can be improved by using the fractional-order derivative instead of the first-order derivative.

\[
EE = \sqrt{(u - u_{GT})^2 + (v - v_{GT})^2}.
\]

\((u, v, 1)\) is the estimated flow vector.

\((u_{GT}, v_{GT}, 1)\) is the ground-truth flow vector.

The accuracy of optical flow estimation algorithms can be improved by using the fractional-order derivative instead of the first-order derivative.
From the table, it can be seen that our model obtains better results than the H–S model for all the image sequences.

It demonstrates the validity of the generalization of differential order.

\[
AE = \arccos \left( \frac{1+u \times u_{GT} + v \times v_{GT}}{\sqrt{1+u \times u + v \times v} \sqrt{1+u_{GT} \times u_{GT} + v_{GT} \times v_{GT}}} \right)
\]

<table>
<thead>
<tr>
<th>Image Sequence</th>
<th>Models</th>
<th>AVAE</th>
<th>SDAE</th>
<th>AVEE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Venus</td>
<td>H–S model</td>
<td>0.1641</td>
<td>0.4176</td>
<td>0.5494</td>
</tr>
<tr>
<td></td>
<td>FOVOF model</td>
<td><strong>0.1561</strong></td>
<td><strong>0.3692</strong></td>
<td><strong>0.5334</strong></td>
</tr>
<tr>
<td>Dimetrodon</td>
<td>H–S model</td>
<td>0.0643</td>
<td>0.0638</td>
<td>0.1908</td>
</tr>
<tr>
<td></td>
<td>FOVOF model</td>
<td><strong>0.0617</strong></td>
<td><strong>0.0630</strong></td>
<td><strong>0.1861</strong></td>
</tr>
<tr>
<td>Hydrangea</td>
<td>H–S model</td>
<td>0.0522</td>
<td>0.1070</td>
<td>0.2967</td>
</tr>
<tr>
<td></td>
<td>FOVOF model</td>
<td><strong>0.0521</strong></td>
<td><strong>0.1063</strong></td>
<td><strong>0.2885</strong></td>
</tr>
<tr>
<td>RubberWhale</td>
<td>H–S model</td>
<td>0.1269</td>
<td>0.2588</td>
<td>0.2268</td>
</tr>
<tr>
<td></td>
<td>FOVOF model</td>
<td><strong>0.1167</strong></td>
<td><strong>0.2439</strong></td>
<td><strong>0.2071</strong></td>
</tr>
<tr>
<td>Grove</td>
<td>H–S model</td>
<td>0.0700</td>
<td>0.1321</td>
<td>0.2919</td>
</tr>
<tr>
<td></td>
<td>FOVOF model</td>
<td><strong>0.0696</strong></td>
<td><strong>0.1307</strong></td>
<td><strong>0.2910</strong></td>
</tr>
<tr>
<td>Urban</td>
<td>H–S model</td>
<td>0.2850</td>
<td>0.6526</td>
<td>1.3944</td>
</tr>
<tr>
<td></td>
<td>FOVOF model</td>
<td><strong>0.1865</strong></td>
<td><strong>0.4813</strong></td>
<td><strong>1.1085</strong></td>
</tr>
</tbody>
</table>
Improved Fractional-Order Variational Optical Flow Model

Model description:

$$\min_{g, w} \int_{\Omega} |D^{\alpha} g| + \int_{\Omega} c(|Dw|)|Dw| + \lambda \| \rho(w, g) \|_1$$  \hspace{1cm} (26)

\( w := (u, v)^T \) is the optic flow field;

\( g \) is the varying illumination;

\( D^{\alpha} := (D^{\alpha}_x, D^{\alpha}_y)^T; \quad |D^{\alpha} u| = \sqrt{(D^{\alpha}_x u)^2 + (D^{\alpha}_y u)^2}. \)

\( c(x) = \frac{1}{1 + \left( \frac{x}{\theta} \right)^2} \)
Discrete Formulation

Discrete model:

\[
\min_{g \in X, w \in Y} \| D^\alpha g \|_1 + \| c(|Dw|) Dw \|_1 + \lambda \| \rho(w, g) \|_1
\]  \hspace{1cm} (27)

\[
\| c(|Dw|) Dw \|_1 = \sum_{i,j} c(|Dw_{ij}|) |Dw_{ij}| \quad \| D^\alpha g \|_1 = \sum_{i,j} |D^\alpha g_{ij}|
\]

\[
\rho(w, g) = I_t + (\nabla I)^T (w - w^0) + \beta g
\]

\(I(x, y, t)\) is the given image sequence;

\((x, y)^T\) denotes the location with a rectangular image domain \(\Omega \in \mathbb{R}\);

\(\nabla I\) is the spatial image gradient;

\(I_t\) is the time image gradient.
Saddle-Point Formulation:

$$\min_{g \in X, w \in Y} \max_{p \in Y, q \in Z} \langle D^\alpha g, p \rangle_Y + \langle c(|Dw|) Dw, q \rangle_Z + \lambda \| \rho(w, g) \|_1 - \delta_P(p) - \delta_Q(q)$$

We define a scalar product in $Y$:

$$\langle a, b \rangle_Y = \sum_{i,j} (a_1 b_1 + a_2 b_2)_{i,j} \quad a = (a_1, a_2) \in Y, \quad b = (b_1, b_2) \in Y$$

We define a scalar product in $Z = Y \times Y$:

$$\langle a, b \rangle_Z = \sum_{i,j} (a_1 b_1 + a_2 b_2 + a_3 b_1 + a_4 b_4)_{i,j}$$

$$a = (a_1, a_2, a_3, a_4) \in Z, \quad b = (b_1, b_2, b_3, b_4) \in Z$$
Saddle-Point Formulation

\[ \delta_P(p) = \begin{cases} 
0, & p \in P \\
+\infty, & \text{others} 
\end{cases} \quad P = \{p \in Y : \|p\|_\infty \leq 1\} \]

\[ \delta_Q(q) = \begin{cases} 
0, & q \in Q \\
+\infty, & \text{others} 
\end{cases} \quad Q = \{q \in Z : \|q\|_\infty \leq 1\} \]
**Numerical Scheme**

**Step 1:**

Let the input image sequence be $I$ and set $n = 0$.

Initialize $w^0, g^0, p^0, q^0, \alpha, \lambda, \beta, \tau, \sigma$ and iteration number $N_{iter}$.

**Step 2:**

\[
\begin{align*}
p^{n+1} &= \frac{\tilde{p}^n}{\max(1, |\tilde{p}^n|)} \\
q^{n+1} &= \frac{\tilde{q}^n}{\max(1, |\tilde{q}^n|)}
\end{align*}
\]

\[
\begin{align*}
\tilde{p}^{n+1} &= p^n + \sigma D^\alpha \tilde{g}^n \\
\tilde{q}^{n+1} &= q^n + \sigma c(|D\tilde{w}^n|) D\tilde{w}^n
\end{align*}
\]
Step 3:

\[ g^{n+1} = \tilde{g}^{n+1} + \begin{cases} 
\tau \lambda \beta, & |\rho^{n+1}| < -\tau \lambda |a|^2 \\
-\tau \lambda \beta, & |\rho^{n+1}| > \tau \lambda |a|^2 \\
-\frac{\rho^{n+1} \beta}{|a|^2}, & |\rho^{n+1}| \leq \tau \lambda |a|^2 
\end{cases} \]

\[ w^{n+1} = \tilde{w}^{n+1} + \begin{cases} 
\tau \lambda \nabla I, & |\rho^{n+1}| < -\tau \lambda |a|^2 \\
-\tau \lambda \nabla I, & |\rho^{n+1}| > \tau \lambda |a|^2 \\
-\frac{\rho^{n+1} \nabla I}{|a|^2}, & |\rho^{n+1}| \leq \tau \lambda |a|^2 
\end{cases} \]

\[ \tilde{g}^{n+1} = g^n - \tau D^{\alpha*} p^{n+1} \quad \tilde{w}^{n+1} = w^n - \tau c(|D^* q^{n+1}|) D^* q^{n+1} \]

\[ |a| = \sqrt{\beta^2 + |\nabla I|^2} \quad \rho^{n+1} = I_t + \nabla I (\tilde{w}^{n+1} - w^0) + \beta \tilde{g}^{n+1} \]
Numerical Scheme

Step 4:
\[ g^{n+1} = 2g^{n+1} - g^n \]
\[ w^{n+1} = 2w^{n+1} - w^n \]

Step 5:
If \( n = N_{iter} \), stop; else, set \( n = n + 1 \) and go to step 2.

Multi-scale approach also is used.
Experiments: Venus

The test images come from [38]
Experiments: Venus

Fractional Order Optical Flow
Demo: Venus

YangQuan Chen, Dali Chen, Dingyu Xue

yangquan.chen@ucmerced.edu; yyqchen@ieee.org
MESA (Mechatronics, Embedded Systems and Automation) LAB,
School of Engineering, University of California, Merced,
5200 N Lake Road, Merced, CA 95343, USA

chendali@ise.neu.edu.cn,
xuedingyu@ise.neu.edu.cn
Information Science and Engineering
Northeastern University
Shenyang 110004, PR China
Experiments: RubberWhale
Experiments: RubberWhale

Fractional Order Optical Flow

Demo: RubberWhale

YangQuan Chen, Dali Chen, Dingyu Xue

yangquan.chen@ucmerced.edu;
yqchen@ieee.org
MESA (Mechatronics, Embedded Systems and Automation) LAB,
School of Engineering, University of California, Merced,
5200 N Lake Road, Merced, CA 95343, USA

chendali@ise.neu.edu.cn,
xuedingyu@ise.neu.edu.cn
Information Science and Engineering
Northeastern University
Shenyang 110004, PR China
Experiments: Hydrangea
Experiments: Hydrangea

Fractional Order Optical Flow

Demo: Hydrangea

YangQuan Chen, Dali Chen, Dingyu Xue

yangquanchen@ucmerced.edu, yqchen@ieee.org
MESA (Mechatronics, Embedded Systems and Automation) LAB,
School of Engineering, University of California, Merced,
5200 N Lake Road, Merced, CA 95343, USA

chendali@ise.neu.edu.cn, xuedingyu@ise.neu.edu.cn
Information Science and Engineering
Northeastern University
Shenyang 110004, PR China
Experiments: Grove
Fractional Order Optical Flow

Demo: Grove

YangQuan Chen, Dali Chen, Dingyu Xue

yangquan.chen@ucmerced.edu;
yqchen@ieee.org
MESA (Mechatronics, Embedded Systems and Automation) Lab,
School of Engineering, University of California, Merced,
5200 N Lake Road, Merced, CA 95343, USA

chendali@ise.neu.edu.cn,
xuedingyu@ise.neu.edu.cn
Information Science and Engineering
Northeastern University
Shenyang 110004, PR China
Experiments: Dimetrodon

The proposed model is able to properly estimate the optical flow.
Experiments: Dimetrodon

Fractional Order Optical Flow

Demo: Dimetrodon

YangQuan Chen, Dali Chen, Dingyu Xue

yangquan.chen@ucmerced.edu;
yqchen@ieee.org
MESA (Mechatronics, Embedded Systems and Automation) LAB,
School of Engineering, University of California, Merced,
5200 N Lake Road, Merced, CA 95343, USA

chendali@ise.neu.edu.cn,
xuedingyu@ise.neu.edu.cn
Information Science and Engineering
Northeastern University
Shenyang 110004, PR China

The proposed model is able to properly estimate the optical flow.
Experiments:

Table 1: AVAE quantitative comparison

<table>
<thead>
<tr>
<th>Model</th>
<th>Venus</th>
<th>Dimetrodon</th>
<th>Hydrangea</th>
<th>RubberWhale</th>
<th>Grove</th>
</tr>
</thead>
<tbody>
<tr>
<td>H-S Model</td>
<td>0.1641</td>
<td>0.0643</td>
<td>0.0522</td>
<td>0.1269</td>
<td>0.0700</td>
</tr>
<tr>
<td>FOVVOF Model</td>
<td>0.1561</td>
<td>0.0617</td>
<td>0.0521</td>
<td>0.1167</td>
<td>0.0696</td>
</tr>
<tr>
<td>IFOVVOF Model($\alpha = 1$)</td>
<td>0.0875</td>
<td>0.0578</td>
<td>0.0401</td>
<td>0.0718</td>
<td>0.0478</td>
</tr>
<tr>
<td>IFOVVOF Model($\alpha = 1.2$)</td>
<td>0.0869</td>
<td>0.0567</td>
<td>0.0393</td>
<td>0.0695</td>
<td>0.0476</td>
</tr>
</tbody>
</table>

- This model obtains better results than the FOVVOF model.
- The generalization of differential order is helpful for improving the accuracy.
Conclusion

- Two fractional-order variational optical flow model are proposed, see in [37];
- Two effective numerical algorithms are proposed;
- They can be combined with the multi-scale approach;
- Improve the accuracy;
- Provide a new tool for motion estimation.
Reference List

Reference List


Take home message:

More optimal image processing can be made possible by using fractional order differentiation and fractional order partial differential equations.

Want to be more optimal? Go fractional calculus!

More info:
http://mechatronics.ucmerced.edu/research/applied-fractional-calculus