

Алгоритм повышения качества смазанных снимков для GPU

А. А. Тихонов¹

¹МГУ, ВМК, каф. ММП

Доклад на спецсеминаре
«Байесовские методы машинного обучения»



Содержание

Проблема смазанных изображений

- Примеры смазанных изображений
- Возможные причины смазанности

Обзор методов восстановления

- Общепринятая модель размытия
- Восстановление с известным ядром

Предлагаемый для реализации метод

Формализованное решение

- Правило Байеса
- Шум
- Ядро
- Скрытое изображение

Оптимизация (EM-алгоритм)

- Приближение скрытого изображения \mathbf{L}
- Вычисление приближения градиента $\partial \mathbf{L} - \Psi$
- Обновление \mathbf{L}
- Нахождение ядра свертки \mathbf{f}

Алгоритм для реализации



Проект выполняется в рамках
«Всероссийской молодежной школы по суперкомпьютерным
технологиям».

Задание предложено компанией NVIDIA.

Кураторы проекта — А. В. Боресков и М. Смирнов



<http://school.hpc-russia.ru/>



Содержание

Проблема смазанных изображений

Примеры смазанных изображений

Возможные причины смазанности

Обзор методов восстановления

Общепринятая модель размытия

Восстановление с известным ядром

Предлагаемый для реализации метод

Формализованное решение

Правило Байеса

Шум

Ядро

Скрытое изображение

Оптимизация (EM-алгоритм)

Приближение скрытого изображения \mathbf{L}

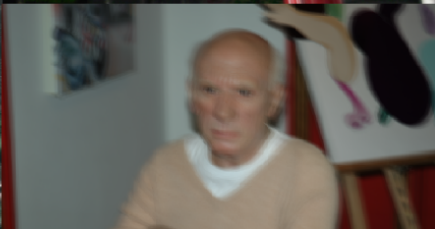
Вычисление приближения градиента $\partial \mathbf{L} - \Psi$

Обновление \mathbf{L}

Нахождение ядра свертки \mathbf{f}

Алгоритм для реализации





Содержание

Проблема смазанных изображений

Примеры смазанных изображений

Возможные причины смазанности

Обзор методов восстановления

Общепринятая модель размытия

Восстановление с известным ядром

Предлагаемый для реализации метод

Формализованное решение

Правило Байеса

Шум

Ядро

Скрытое изображение

Оптимизация (EM-алгоритм)

Приближение скрытого изображения \mathbf{L}

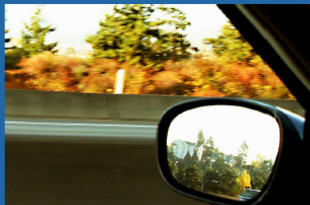
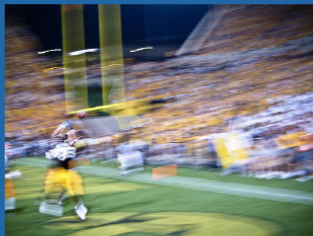
Вычисление приближения градиента $\partial \mathbf{L} - \Psi$

Обновление \mathbf{L}

Нахождение ядра свертки \mathbf{f}

Алгоритм для реализации





Причинами смазанности могут выступать различные факторы:

- ▶ Движение камеры в процессе съемки изображения;
- ▶ Съемка на длинной выдержке, когда сцена сама претерпевает изменения;
- ▶ Расфокусированность оптики;
- ▶ Использование широкоугольных объективов;
- ▶ Атмосферная турбулентность;
- ▶ Съемка на короткой выдержке, что не позволяет захватить достаточно фотонов;
- ▶ Рассеяние света в конфокальной микроскопии;



Содержание

Проблема смазанных изображений

Примеры смазанных изображений

Возможные причины смазанности

Обзор методов восстановления

Общепринятая модель размытия

Восстановление с известным ядром

Предлагаемый для реализации метод

Формализованное решение

Правило Байеса

Шум

Ядро

Скрытое изображение

Оптимизация (EM-алгоритм)

Приближение скрытого изображения \mathbf{L}

Вычисление приближения градиента $\partial \mathbf{L} - \Psi$

Обновление \mathbf{L}

Нахождение ядра свертки \mathbf{f}

Алгоритм для реализации



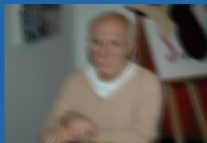
Общепринятая модель размытия — свертка

$$\mathbf{l} = \mathbf{L} \otimes \mathbf{f} + \mathbf{n};$$

\mathbf{n} — шум

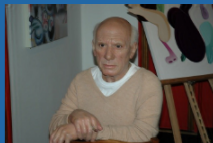


Blur Model



Blurred image I

=



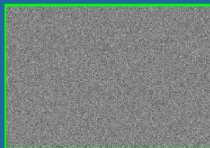
Sharp image L

\otimes



Blur kernel f

+



Camera Noise n



Содержание

Проблема смазанных изображений

- Примеры смазанных изображений
- Возможные причины смазанности

Обзор методов восстановления

- Общепринятая модель размытия
- Восстановление с известным ядром**

Предлагаемый для реализации метод

- Формализованное решение
 - Правило Байеса
 - Шум
 - Ядро
 - Скрытое изображение

Оптимизация (EM-алгоритм)

- Приближение скрытого изображения \mathbf{L}
- Вычисление приближения градиента $\partial \mathbf{L} - \Psi$
- Обновление \mathbf{L}
- Нахождение ядра свертки \mathbf{f}

Алгоритм для реализации





Non-blind deconvolution



Blind deconvolution



Фильтр Винера из обработки сигналов

$$y(t) = h(t) \otimes x(t) + v(t);$$

$x(t)$ — входной сигнал в момент времени t

$h(t)$ — известный импульсный отклик линейной инвариантной к сдвигу во времени системы

$v(t)$ — шум, не зависящий от $x(t)$

$y(t)$ — наблюдаемый сигнал



Цель — найти такую $g(t)$, что можно вычислить приближение $x(t)$:

$$\hat{x}(t) = g(t) \otimes y(t);$$

Причем ищем такое $\hat{x}(t)$, которое минимизирует матожидание квадрата отклонения от $x(t)$.



Фильтр Винера дает такой ответ:

$$G(f) = \frac{H^*(f)S(f)}{|H(f)|^2S(f) + N(f)};$$

Где:

- $G(f)$ и $H(f)$ есть образы Фурье соответственно g и h , где f
- некоторая частота
- $S(f)$ — средняя спектральная плотность входного сигнала $x(t)$
- $N(f)$ — средняя спектральная плотность шума $v(t)$
- под знаком $*$ понимается комплексное сопряжение



Смысл работы фильтра Винера:

$$G(f) = \frac{1}{H(f)} \left[\frac{|H(f)|^2}{|H(f)|^2 + \frac{N(f)}{S(f)}} \right];$$

Неизвестные спектральные плотности на практике часто удается оценить/приблизить.



Хотим минимизировать среднеквадратичную ошибку:

$$\epsilon(f) = \mathbb{E} \left| X(f) - \hat{X}(f) \right|^2$$

$$\epsilon(f) = \mathbb{E} |X(f) - G(f)Y(f)|^2$$

$$= \mathbb{E} |X(f) - G(f)[H(f)X(f) + V(f)]|^2$$

$$= \mathbb{E} | [1 - G(f)H(f)] X(f) - G(f)V(f) |^2$$

$$\epsilon(f) = \left[1 - G(f)H(f) \right] \left[1 - G(f)H(f) \right]^* \mathbb{E} |X(f)|^2$$

$$+ \left[1 - G(f)H(f) \right] G^*(f) \mathbb{E} \left\{ X(f)V^*(f) \right\}$$

$$+ G(f) \left[1 - G(f)H(f) \right]^* \mathbb{E} \left\{ V(f)X^*(f) \right\}$$

$$+ G(f)G^*(f) \mathbb{E} |V(f)|^2$$

Шум не зависит от сигнала:

$$\mathbb{E} \left\{ X(f)V^*(f) \right\} = \mathbb{E} \left\{ V(f)X^*(f) \right\} = 0$$

Спектральные плотности определим так:

$$S(f) = \mathbb{E} |X(f)|^2 \quad N(f) = \mathbb{E} |V(f)|^2$$



Итого:

$$\begin{aligned}\epsilon(f) = & \left[1 - G(f)H(f)\right] \left[1 - G(f)H(f)\right]^* S(f) \\ & + G(f)G^*(f)N(f)\end{aligned}$$

Чтобы ошибка была минимальная, дифференцируем по $G(f)$ и приравниваем к нулю.

$$\frac{d\epsilon(f)}{dG(f)} = G^*(f)N(f) - H(f) \left[1 - G(f)H(f)\right]^* S(f) = 0;$$

$$G(f) = \frac{H^*(f)S(f)}{|H(f)|^2 S(f) + N(f)};$$



Пример работы фильтра Винера



$$G(f) = \frac{H^*(f)}{|H(f)|^2 + r(f)} G_0(f);$$

Существуют и другие аналогичные фильтры. По сути есть байесовская регуляризация решения. Регуляризация необходима вследствие «некорректности» задачи восстановления. В [5] подробно излагается, что тут возможны два способа регуляризации — согласно байесовскому выводу и по Тихонову. Оба вводят априорные знания: байесовская (вероятностная) — о плотности распределения всех изображений-кандидатов, давая приоритет более вероятным, по Тихонову (детерминисткая) — некий специальный энергетический функционал (к примеру, $a\|u\|^2$).



Алгоритм Люси-Ричардсона

Пиксели наблюдаемого изображения могут быть выражены с помощью ядра и скрытого изображения:

$$d_i = \sum_j p_{ij} u_j$$

где p_{ij} есть пиксель ядра, u_j — яркость j -го пикселя скрытого изображения, а d_i — наблюдаемая яркость i -го пикселя. Также сделано предположение о пуассоновском ($f(k; \lambda) = \frac{\lambda^k e^{-\lambda}}{k!}$) распределении пикселей u_j , что корректно для фотонного шума.



Идея метода в попытке найти наиболее вероятные значения u_j , имея значения d_i и p_{ij} . Это дает уравнения для u_j , которые решаются итеративно по формуле:

$$u_j^{(t+1)} = u_j^{(t)} \sum_i \frac{d_i}{c_i} p_{ij}$$

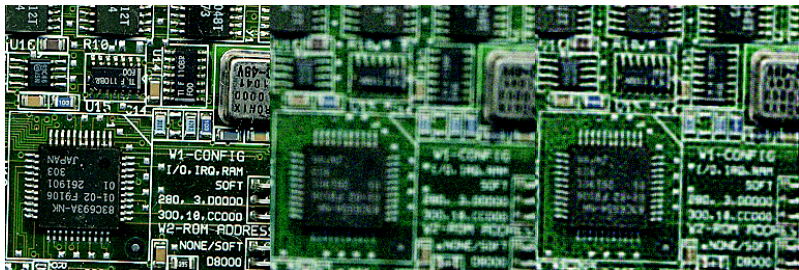
Здесь $c_i = \sum_j p_{ij} u_j^{(t)}$.

Эмпирически показано [5], что если метод сходится, то он сходится к ответу с наибольшим правдоподобием.

Если ядро не так тривиально, имеет один и более неизвестный параметр, метод перестает работать.



Пример работы метода Люси-Ричардсона:



На практике ядро неизвестно почти всегда. Поэтому давно идут попытки оценить его так или иначе.

Придуманы аппаратные устройства, позволяющие оценить ядро (два отдельных кадра, кадры с разной фокусировкой, освещенностью, парные камеры, фото + видео и т. д.), многие программные методы также требуют дополнительной.



Пример аппаратного устройства [4]:



Пример его работы:



Еще примеры:

Previous Work (1)



SIGGRAPH2008

Hardware solutions:



[Ben-Ezra and Nayar 2004]



[Levin et al. 2008]



[Raskar et al. 2006] 4



Previous Work (2)

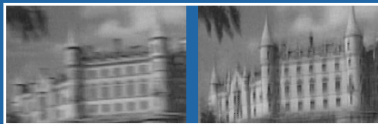


SIGGRAPH2008

Multi-frame solutions:



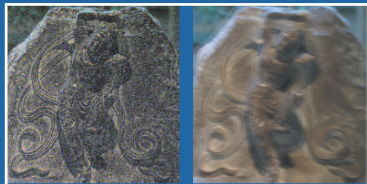
[Jia et al. 2004]



[Rav-Acha and Peleg 2005]



[Petschnigg et al. 2004]



[Yuan et al. 2007]

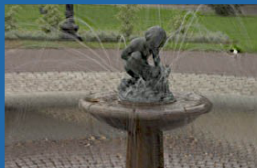


Previous Work (3)

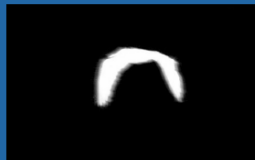


SIGGRAPH2008

Single image solutions:



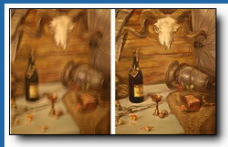
[Fergus et al. 2006]



[Jia 2007]



[Levin et al. 2007]



[Yuan et al. 2008]



Все это неудобно, и обычно неприменимо из-за дороговизны и сложности. Поэтому важной задачей является построение хорошего универсального алгоритма, который сможет оценить ядро и затем восстановить изображение по одному изображению.



Один из лучших с точки зрения качества алгоритм такого рода описан в [1]. Он представляет собой двухшаговый итерационный процесс, одна из итераций которого восстанавливает картинку для некоторого приближения ядра, а другая производит уточнение этого ядра по полученному на первом шаге снимку.

Этот способ представляет собой частный случай известного EM-алгоритма (см. [1]).



Содержание

Проблема смазанных изображений

- Примеры смазанных изображений
- Возможные причины смазанности

Обзор методов восстановления

- Общепринятая модель размытия
- Восстановление с известным ядром

Предлагаемый для реализации метод

Формализованное решение

- Правило Байеса
- Шум
- Ядро
- Скрытое изображение

Оптимизация (EM-алгоритм)

- Приближение скрытого изображения \mathbf{L}
- Вычисление приближения градиента $\partial \mathbf{L} - \Psi$
- Обновление \mathbf{L}
- Нахождение ядра свертки \mathbf{f}

Алгоритм для реализации



По формуле Байеса:

$$p(\mathbf{L}, \mathbf{f}|\mathbf{I}) \propto p(\mathbf{I}|\mathbf{L}, \mathbf{f})p(\mathbf{L})p(\mathbf{f});$$

Теперь рассмотрим каждый множитель.



Шум \mathbf{n} — одинаковое распределение (нормальное) для всех пикселей изображения.

$$p(\mathbf{I}|\mathbf{L}, \mathbf{f}) = \prod_{\partial^* \in \Theta} \prod_i \mathcal{N}(\partial^* n_i | 0, \zeta_{\kappa(\partial^*)}) = \prod_{\partial^* \in \Theta} \prod_i \mathcal{N}(\partial^* I_i | I_i^c, \zeta_{\kappa(\partial^*)});$$

Θ — множество производных ($\Theta = (\partial^0, \partial_x, \partial_y, \partial_{xx}, \partial_{xy}, \partial_{yy})$), I_i^c — i -й пиксель изображения $\mathbf{I}^c = \mathbf{L} \otimes \mathbf{f}$.



Априорная вероятность конфигурации ядра свертки $p(\mathbf{f})$ выбрана таким образом, чтобы оно было разреженным, соответственно движению камеры:

$$p(\mathbf{f}) = \prod_j e^{-\tau f_j};$$

Здесь τ — параметр скорости.



Разложим правдоподобие в произведение локальной и глобальной компонент:

$$p(\mathbf{L}) = p_g(\mathbf{L})p_l(\mathbf{L});$$



Авторы предлагают простую формулу, аппроксимирующую функцию распределения плотностей в реальных изображениях, которая подсчитана статистически. В предыдущих работах по данной теме были проведены попытки приблизить такую функцию смесью гауссиан в каждой точке изображения, но это выливается (при логарифмировании) в необходимость оптимизировать логарифм от суммы экспоненциальных зависимостей, что на практике является сложной и неустойчивой задачей.



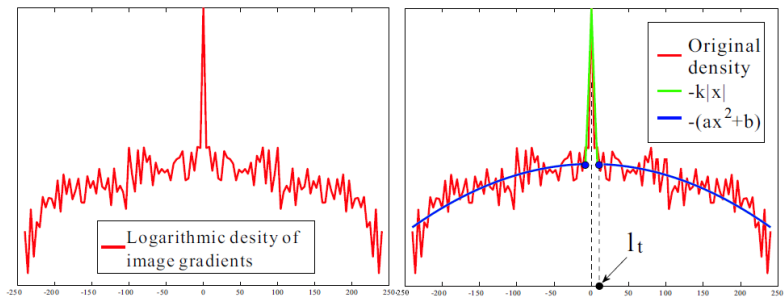


Рис.: График распределения градиентов в реальных изображениях и предлагаемая аппроксимация



Предлагаемая аппроксимирующая формула крайне проста:

$$\Phi(x) = \begin{cases} -k|x|, & \text{если } x < l_t; \\ -(ax^2 + b), & \text{если } x \geq l_t. \end{cases};$$

Здесь k, a, b, l_t — некоторые коэффициенты. С использованием этого приближения глобальная компонента ищется по формуле:

$$p_g(\mathbf{L}) \propto \prod_i e^{\Phi(\partial L_i)};$$



Локальная компонента вводится только для тех областей изображения, где локальная дисперсия (которая считается в окне того же размера, что и ядро свертки) небольшая. В этих областях градиенты исходного и размытого изображений не должны сильно отличаться. Поэтому предлагается использовать нормальное распределение:

$$p_I(\mathbf{L}) = \prod_{i \in \Omega} \mathcal{N}(\partial_x L_i - \partial_x I_i | 0, \sigma_1) \mathcal{N}(\partial_y L_i - \partial_y I_i | 0, \sigma_1);$$

Здесь за Ω обозначены точки изображения с локальной дисперсией менее некоторой константы.



Содержание

Проблема смазанных изображений

- Примеры смазанных изображений
- Возможные причины смазанности

Обзор методов восстановления

- Общепринятая модель размытия
- Восстановление с известным ядром

Предлагаемый для реализации метод

- Формализованное решение
 - Правило Байеса
 - Шум
 - Ядро
 - Скрытое изображение

Оптимизация (EM-алгоритм)

- Приближение скрытого изображения \mathbf{L}
- Вычисление приближения градиента $\partial \mathbf{L} - \Psi$
- Обновление \mathbf{L}
- Нахождение ядра свертки \mathbf{f}

Алгоритм для реализации



$$E(\mathbf{L}, \mathbf{f}) = -\log p(\mathbf{L}, \mathbf{f}|\mathbf{I});$$

$$E(\mathbf{L}, \mathbf{f}) \propto \left(\sum_{\partial^* \in \Omega} w_{\kappa(\partial^*)} \|\partial^* \mathbf{L} \otimes \mathbf{f} - \partial^* \mathbf{I}\|_2^2 \right) + \lambda_1 \|\Phi(\partial_x \mathbf{L}) + \Phi(\partial_y \mathbf{L})\|_1 + \\ + \lambda_2 \left(\|\partial_x \mathbf{L} - \partial_x \mathbf{I}\|_2^2 \circ \mathbf{M} + \|\partial_y \mathbf{L} - \partial_y \mathbf{I}\|_2^2 \circ \mathbf{M} \right) + \|\mathbf{f}\|_1; \quad (1)$$

Здесь \circ обозначает поэлементную операцию умножения. \mathbf{M} — маска, выделяющая в изображении точки с небольшой локальной дисперсией (из Ω). Параметры были для простоты переобозначены, а именно:

$$w_{\kappa(\partial^*)} = \frac{1}{\zeta_{\kappa(\partial^*)}^2 \tau}, \quad \lambda_1 = \frac{1}{\tau}, \quad \lambda_2 = \frac{1}{\sigma_1^2 \tau};$$

Прямая оптимизация этого функционала сложна, сходимость медленная. Авторы [1] предлагают свой вариант, основанный на пошаговой оптимизации \mathbf{L} , а затем \mathbf{f} .



Зафиксируем некоторое приближение ядра \mathbf{f} :

$$E_{\mathbf{L}}(\mathbf{L}, \mathbf{f}) = \left(\sum_{\partial^* \in \Omega} w_{\kappa(\partial^*)} \|\partial^* \mathbf{L} \otimes \mathbf{f} - \partial^* \mathbf{I}\|_2^2 \right) + \lambda_1 \|\Phi(\partial_x \mathbf{L}) + \Phi(\partial_y \mathbf{L})\|_1 + \\ + \lambda_2 \left(\|\partial_x \mathbf{L} - \partial_x \mathbf{I}\|_2^2 \circ \mathbf{M} + \|\partial_y \mathbf{L} - \partial_y \mathbf{I}\|_2^2 \circ \mathbf{M} \right);$$

Это сильно невыпуклая функция от многих тысяч переменных, Прямая оптимизация очень трудоемка (а это нужно делать на каждой итерации).



Введем дополнительные переменные $\Psi = (\Psi_x, \Psi_y)$, которые будут аппроксимировать $\partial \mathbf{L}$, таким образом, каждой паре градиентов пикселя i — $(\partial_x L_i, \partial_y L_i) \in \partial \mathbf{L}$ — будет соответствовать пара $(\psi_{i,x}, \psi_{i,y}) \in \Psi$.

При фиксированном \mathbf{L} можно найти такие Ψ , которые будут минимизировать энергию. То есть, вместо подбора всего изображения, подберем лишь его градиент (влияет лишь на часть выражения), а затем по найденным градиентам обновим \mathbf{L} . То есть, перешли к минимизации:

$$\begin{aligned}
 E_{\mathbf{L}}(\mathbf{L}', \mathbf{f}) = & \left(\sum_{\partial^* \in \Omega} w_{\kappa(\partial^*)} \|\partial^* \mathbf{L} \otimes \mathbf{f} - \partial^* \mathbf{I}\|_2^2 \right) + \lambda_1 \|\Phi(\Psi_x) + \Phi(\Psi_y)\|_1 + \\
 & + \lambda_2 \left(\|\Psi_x - \partial_x \mathbf{I}\|_2^2 \circ \mathbf{M} + \|\Psi_y - \partial_y \mathbf{I}\|_2^2 \circ \mathbf{M} \right) + \\
 & + \gamma \left(\|\Psi_x - \partial_x \mathbf{L}\|_2^2 + \|\Psi_y - \partial_y \mathbf{L}\|_2^2 \right);
 \end{aligned}$$



С увеличением номера итерации параметр регуляризации по отклонению градиентов γ также будет расти, и в конечном итоге Ψ будет практически совпадать с $\partial\mathbf{L}$, что означает, что минимизация $E_{\mathbf{L}}'$ через несколько шагов процесса будет эквивалентна минимизации $E_{\mathbf{L}}$.
Распишем оба шага подробнее.



При фиксированном \mathbf{L} , а значит, и $\partial\mathbf{L}$, остается оптимизировать E_Ψ :

$$E_\Psi(\mathbf{L}, \mathbf{f}) = \lambda_1 \|\Phi(\Psi_x) + \Phi(\Psi_y)\|_1 + \lambda_2 \|\Psi_x - \partial_x \mathbf{I}\|_2^2 \circ \mathbf{M} + \lambda_2 \|\Psi_y - \partial_y \mathbf{I}\|_2^2 \circ \mathbf{M} + \gamma \|\Psi_x - \partial_x \mathbf{L}\|_2^2 + \gamma \|\Psi_y - \partial_y \mathbf{L}\|_2^2; \quad (2)$$

Можно произвести дальнейшее разделение по пикселям и координатам x, y , получим, что для $\nu \in \{x, y\}$ нужно минимизировать:

$$E_{\psi_{i,\nu}} = \lambda_1 |\Phi(\psi_{i,\nu})| + \lambda_2 m_i (\Psi_{i,\nu} - \partial_\nu l_i)^2 + \gamma (\Psi_{i,\nu} - \partial_\nu L_i)^2;$$



Каждая компонента $E_{\psi_{i,\nu}}$ содержит лишь одну переменную $\psi_{i,\nu}$, поэтому они легко могут быть оптимизированы, а результаты собраны вместе.

Φ состоит из 4-х выпуклых частей, каждая может быть оптимизирована отдельно, а затем лучший результат взят для определения глобально минимального Ψ .



При фиксированном Ψ , будем обновлять \mathbf{L} , минимизируя по нему следующее выражение:

$$E'_{\mathbf{L}} = \left(\sum_{\partial^* \in \Omega} w_{\kappa(\partial^*)} \|\partial^* \mathbf{L} \otimes \mathbf{f} - \partial^* \mathbf{I}\|_2^2 \right) + \\ + \gamma \|\Psi_x - \partial_x \mathbf{L}\|_2^2 + \gamma \|\Psi_y - \partial_y \mathbf{L}\|_2^2;$$

Так как здесь присутствует свертка, выгодно перейти к работе с образами Фурье. Если обозначить оператор преобразования Фурье за \mathcal{F} , а за \mathcal{F}^{-1} — ему обратный, то получим:

$$E'_{\mathcal{F}(\mathbf{L})} = \left(\sum_{\partial^* \in \Omega} w_{\kappa(\partial^*)} \|\mathcal{F}(\mathbf{L}) \circ \mathcal{F}(\mathbf{f}) \circ \mathcal{F}(\partial^*) - \mathbf{F}(\mathbf{I}) \circ \mathcal{F}(\partial^*)\|_2^2 \right) + \\ + \gamma \|\mathcal{F}(\Psi_x) - \mathcal{F}(\mathbf{L}) \circ \mathcal{F}(\partial_x)\|_2^2 + \gamma \|\mathcal{F}(\Psi_y) - \mathcal{F}(\mathbf{L}) \circ \mathcal{F}(\partial_y)\|_2^2;$$



Ссылаясь на формулу Планшереля ([2]), авторы [1] переходят от минимизации $E_{\mathbf{L}}'$ к эквивалентной задаче минимизации $E'_{\mathcal{F}(\mathbf{L})}$. Таким образом, искомое значение затем находится с помощью обратного преобразования Фурье:

$$\mathbf{L}^* = \mathcal{F}^{-1}(\arg \min_{\mathcal{F}(\mathbf{L})} E_{\mathcal{F}(\mathbf{L})}')$$



Теперь минимальное значение такой энергии может быть получено, если приравнять к нулю его производную $\partial E'_{\mathcal{F}(\mathbf{L})} / \partial \mathcal{F}_{\mathbf{L}}$, это можно сделать в явном виде, откуда:

$$\mathbf{L}^* = \mathcal{F}^{-1} \left(\frac{\overline{\mathcal{F}(\mathbf{f})} \circ \mathcal{F}(\mathbf{l}) \circ \Delta + \gamma \overline{\mathcal{F}(\partial_x)} \circ \mathcal{F}(\Psi_x) + \gamma \overline{\mathcal{F}(\partial_y)} \circ \mathcal{F}(\Psi_y)}{\overline{\mathcal{F}(\mathbf{f})} \circ \mathcal{F}(\mathbf{f}) \circ \Delta + \gamma \overline{\mathcal{F}(\partial_x)} \circ \mathcal{F}(\partial_x) + \gamma \overline{\mathcal{F}(\partial_y)} \circ \mathcal{F}(\partial_y)} \right); \quad (3)$$

Тут $\Delta = \sum_{\partial^* \in \Theta} w_{\kappa(\partial^*)} \overline{\mathcal{F}(\partial^*)} \circ \mathcal{F}(\partial^*)$, $\overline{(\cdot)}$ обозначает оператор комплексного сопряжения, и деление производится поэлементно.



Будем оптимизировать энергию при фиксированном \mathbf{L} :

$$E(\mathbf{f}) = \left(\sum_{\partial^* \in \Omega} w_{\kappa(\partial^*)} \|\partial^* \mathbf{L} \otimes \mathbf{f} - \partial^* \mathbf{I}\|_2^2 \right) + \|\mathbf{f}\|_1; \quad (4)$$

Авторы алгоритма ссылаются на работу [3], в которой функционал такого вида успешно оптимизируется с помощью специального interior point method.





Содержание

Проблема смазанных изображений

- Примеры смазанных изображений
- Возможные причины смазанности

Обзор методов восстановления

- Общепринятая модель размытия
- Восстановление с известным ядром

Предлагаемый для реализации метод

- Формализованное решение
 - Правило Байеса
 - Шум
 - Ядро
 - Скрытое изображение

Оптимизация (EM-алгоритм)

- Приближение скрытого изображения \mathbf{L}
- Вычисление приближения градиента $\partial \mathbf{L} - \Psi$
- Обновление \mathbf{L}
- Нахождение ядра свертки \mathbf{f}

Алгоритм для реализации



Схема полученного алгоритма:

Вход:

\mathbf{I} — размытое изображение, \mathbf{f} — начальное приближение ядра;

Выход:

\mathbf{L} — искомое четкое изображение, \mathbf{f} — исходное ядро размытия;

$\mathbf{L} \leftarrow \mathbf{I}$; // инициализация скрытого изображения
наблюдаемым;

оптимизация \mathbf{L} и \mathbf{f} :

повторять

оптимизация \mathbf{L} :

повторять

Обновить Ψ , минимизируя (2);

Вычислить \mathbf{L} согласно (3);

пока $\|\Delta \mathbf{L}\|_2 < 1 \times 10^{-5}$ и $\|\Delta \Psi\|_2 < 1 \times 10^{-5}$;

Обновить \mathbf{f} , минимизируя (4);

пока $\|\Delta \mathbf{f}\|_2 < 1 \times 10^{-5}$ или максимальное число итераций
завершено;








Итого два итерационных процесса — внутренний (вложенный), чередование вычисления Ψ и \mathbf{L} , и внешний, вычисление очередного приближения скрытой картинки \mathbf{L} и на его основе уточнение ядра \mathbf{f} .



Литература

-  Shan Q., Jia J., Agarwala A. — High-quality Motion Deblurring from a Single Image.// ACM Transactions on Graphics (SIGGRAPH), 2008.
-  Tony F. C., Shen J. — Theory and Computation of Variational Image Deblurring, 2005.
-  Гонсалес Р., Вудс Р. — Цифровая обработка изображений.// М.: Техносфера, 2005.
-  Richardson W. H. — Bayesian-Based Iterative Method of Image Restoration.// Journal of the Optical Society of America, vol. 62, 1, 1972.
-  Tony F. C., Shen J. — Theory and Computation of Variational Image Deblurring, 2005.



-  Dempster A.P., Laird N.M., Rubin D.B. — Maximum likelihood from incomplete data via the EM algorithm.// J. Royal Stat. Soc. Ser. B, 39 (1), pp. 1–38, 1977
-  de Branges L., — Hilbert spaces of entire functions.// Prentice-Hall, N. Y., 1968
-  Kim S.-J., Koh K., Lustig M., Boyd S. — An efficient method for compressed sensing.// ICIP proceedings, 2007
-  Joshi, N., Kang, S. B., Zitnick, C. L., and Szeliski, R. 2010. Image deblurring using inertial measurement sensors. ACM Trans. Graph. 29, 4 (Jul. 2010)
-  Shepp, L. A.; Vardi, Y. (1982), "Maximum Likelihood Reconstruction for Emission Tomography IEEE Transactions on Medical Imaging



Спасибо за внимание

