THE MOREAU ENVELOPE APPROACH FOR THE_L1_TV_IMAGE_DENOISING_MODEL-Chen-Shen-Xu-Zeng-2014

По факту тут приведены разные варианты реализации l1_tv-модели

$$\min_u \{\lambda\|u - x\|_1 + \|u\|_{\mathrm{TV}}\}$$

u : Это переменная, которую мы оптимизируем. В контексте обработки изображений u может представлять восстановленное или очищенное изображение.

x : Это исходное изображение или сигнал, который может быть зашумленным или искаженным.

$\lambda > 0$: Параметр баланса между двумя терминами. Он управляет относительным весом каждого слагаемого.

Использование L1-нормы вместо L2-нормы (как в MSE) делает эту модель более устойчивой к выбросам и позволяет сохранять резкие переходы (например, края объектов на изображении).

$$\|u\|_{\mathrm{TV}} = \sum_{i,j} \sqrt{\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2}$$

Это **Total Variation (TV)-норма** изображения u. Она измеряет сумму градиентов (или изменений интенсивности) пикселей в изображении. В дискретном случае она может быть записана как:

$$\|u\|_{\mathrm{TV}} = \sum_{i,j} (|u_{i+1,j} - u_{i,j}| + |u_{i,j+1} - u_{i,j}|)$$

где $u_{i,j}$ — интенсивность пикселя в точке (i, j).

TV-норма способствует получению разряженного градиента , т. е. изображения с резкими границами и минимальными изменениями внутри однородных областей.

Это особенно полезно для задач, где важно сохранить края объектов (например, в задачах сегментации или восстановления изображений).

Если u имеет резкие переходы (например, края), то TV-норма остается небольшой.

Если u содержит много мелких деталей или шума, то TV-норма увеличивается.

- **Первый термин** ($\lambda\|u - x\|_1$) стремится сделать u похожим на x.
- **Второй термин** ($\|u\|_{\mathrm{TV}}$) стремится сделать u гладким и с резкими границами.

Параметр λ регулирует баланс между этими двумя целями:

- Большое λ: Приоритет отдается точному приближению u к x, но возможны "мягкие" края.
- Маленькое λ: Приоритет отдается гладкости и резким границам, но u может отклоняться от x.

Они рассматривают данную модель, применяя функцию

$$\operatorname{env}_\beta \psi(z) := \min_{y \in \mathbb{R}^m} \left\{ \frac{1}{2\beta} \|y - z\|_2^2 + \psi(y) \right\}$$

Объяснение функции:

Эта формула описывает **обобщенную проекцию** или **proximal оператор** , который часто используется в оптимизации и машинном обучении. Она называется **envelope function** (функция оболочки) или **Moreau envelope** .

Что делает эта функция?

Функция $\operatorname{env}_\beta \psi(z)$ минимизирует сумму двух терминов:

1. **Квадратичное расстояние** : $\frac{1}{2\beta} \|y - z\|_2^2$, которое измеряет близость точки y к точке z.
2. **Целевая функция** : $\psi(y)$, которая представляет собой некоторую целевую функцию, которую мы хотим минимизировать.

Минимизация этой суммы позволяет найти точку y, которая балансирует между близостью к z и минимизацией $\psi(y)$.

Переменные и их значения:

1. $\operatorname{env}_\beta \psi(z)$:
   - Это результат применения оператора оболочки к функции ψ с параметром β в точке z.
   - Это значение является минимальным значением выражения внутри скобок при оптимальном выборе y.
2. $\beta > 0$:
   - Параметр β регулирует "степень гладкости" функции $\operatorname{env}_\beta \psi(z)$. Чем больше β, тем более гладкой становится функция $\operatorname{env}_\beta \psi(z)$.
   - Он также влияет на вес квадратичного слагаемого по сравнению с $\psi(y)$.
3. $z \in \mathbb{R}^m$:
   - Точка, относительно которой производится минимизация. Это входная точка для функции $\operatorname{env}_\beta \psi(z)$.
4. $y \in \mathbb{R}^m$:
   - Переменная, по которой происходит минимизация. Это точка, которая выбирается так, чтобы минимизировать выражение $\frac{1}{2\beta} \|y - z\|_2^2 + \psi(y)$.

5. $\|y - z\|_2^2$:

   - Квадрат евклидова расстояния между точками y и z. Это измеряет, насколько близка точка y к точке z.

6. $\psi(y)$:

   - Целевая функция, зависящая от y. Она может быть любой выпуклой функцией, которую мы хотим минимизировать. Например, это может быть функция потерь, штраф или регуляризация.

Геометрическая интерпретация:

- Функция $\frac{1}{2\beta}\|y - z\|_2^2$ представляет собой параболоид, который стремится "тянуть" y к z.
- Функция $\psi(y)$ представляет собой целевую поверхность, которую нужно минимизировать.
- Минимизация их суммы позволяет найти компромисс между близостью к z и минимизацией $\psi(y)$.

Применяя данную функцию к исходной, они получили 5 вариантов

МОДЕЛЬ	ТЕРМ ТОЧНОСТИ	РЕГУЛЯРИЗАЦИОННЫЙ ТЕРМ	МОДЕЛЬ
1	$\|u - x\|_1$	$\varphi(Bu)$	$\min_u \{\lambda\|u - x\|_1 + \varphi(Bu)\}$
2	$\mathrm{env}_{\beta\|\cdot\|_1}(u - x)$	$\varphi(Bu)$	$\min_u \left\{\lambda\mathrm{env}_{\beta\|\cdot\|_1}(u - x) + \varphi(Bu)\right\}$
3	$\|u - x\|_1$	$\mathrm{env}_{\beta\varphi}(Bu)$	$\min_u \{\lambda\|u - x\|_1 + \mathrm{env}_{\beta\varphi}(Bu)\}$
4	$\mathrm{env}_{\beta\|\cdot\|_1}(u - x)$	$\mathrm{env}_{\beta\varphi}(Bu)$	$\min_u \left\{\lambda\mathrm{env}_{\beta\|\cdot\|_1}(u - x) + \mathrm{env}_{\beta\varphi}(Bu)\right\}$
5	$\|u - x\|_1$	$\mathrm{env}_{\beta\varphi\circ B}(Bu)$	$\min_u \{\lambda\|u - x\|_1 + \mathrm{env}_{\beta\varphi\circ B}(Bu)\}$
6	$\mathrm{env}_{\beta\|\cdot\|_1}(u - x)$	$\mathrm{env}_{\beta\varphi\circ B}(Bu)$	$\min_u \left\{\lambda\mathrm{env}_{\beta\|\cdot\|_1}(u - x) + \mathrm{env}_{\beta\varphi\circ B}(Bu)\right\}$

Самой эффективной из них оказалась модель 3, тк она наиболее предпочтительная для задачи удаления импульсного шума (шум соль-перец).

Модель 3 основана на следующей формуле:

$$\min_u \{\lambda\|u - x\|_1 + \mathrm{env}_{\beta\varphi}(Bu)\},$$

где:

- x — зашумленное изображение.
- u — восстанавливаемое изображение.
- λ — параметр баланса между термом точности и регуляризатором.

- $\|u - x\|_1$ — терм точности, представляющий ℓ^1-норму разницы между исходным изображением и зашумленным.
- $\mathrm{env}_{\beta\varphi}(Bu)$ — сглаженная полная вариация, где:
    - B — матрица разностей, описывающая полную вариацию изображения.
    - φ — функция, связанная с полной вариацией.
    - $\mathrm{env}_{\beta\varphi}$ — оболочка Морана функции φ, которая делает её дифференцируемой.

Для решения модели 3 были использованы два основных подхода:

Алгоритм 1 : Ускоренный метод Гаусса-Зейделя.

Algorithm 1 Proximity algorithm for Models 1-4 accelerated by GS iteration

Given: A noisy image x in $\mathbb{R}^d$; $\lambda > 0$, $\beta > 0$, $\gamma > 0$, $\sigma > 0$ such that $\frac{\sigma}{\gamma} < \frac{1}{8}$
Initialization: $u^0 = x$, $v^0 = 0$, $b^0 = 0$
repeat
(a) Update the components of u^{k+1} according to equation (38)
(b) $v^{k+1} \leftarrow \mathrm{prox}_{\frac{1}{\sigma}R}(b^k + Bu^{k+1})$
(c) $b^{k+1} \leftarrow b^k + Bu^{k+1} - v^{k+1}$
until u^{k+1} converges or satisfies a stopping criteria.

(перевод с англ.)

Дано: Зашифрованное изображение $x \in \mathbb{R}^d$; $\lambda > 0$, $\beta > 0$, $\gamma > 0$, $\sigma > 0$ такие, что $\frac{\sigma}{\gamma} < \frac{1}{8}$.

Инициализация: $u^0 = x$, $v^0 = 0$, $b^0 = 0$.

Повторять:

1. Обновить компоненты u^{k+1} согласно уравнению (38).
2. $v^{k+1} \leftarrow \mathrm{prox}_{\frac{1}{\gamma}R}(b^k + Bu^{k+1})$.
3. $b^{k+1} \leftarrow b^k + Bu^{k+1} - v^{k+1}$.

Пока u^{k+1} сходится или удовлетворяет критерию остановки.

Он показывает быструю сходимость при небольших значениях β (меньше 10).

Алгоритм 6 : Комбинация FISTA и метода Гаусса-Зейделя.

Этот алгоритм применяет FISTA для ускорения сходимости итерационного процесса.

Он особенно эффективен при больших значениях β (от 10 до 20), так как FISTA способствует более быстрой минимизации целевой функции.

Algorithm 6 The proximity algorithm for Model 3 or Model 4 accelerated by FISTA and GS iteration

Given: A noisy image x in $\mathbb{R}^d$; $\lambda > 0$, $\beta > 0$, $\gamma > 0$ such that $\frac{1}{\gamma\beta} < \frac{1}{4}$

Initialization: $y^1 = u^0 = x$, $u^{-1} = 0$, $t^1 = 1$

repeat

(a) Update the components of y^{k+1} according to equation (55).

(b) $u^k = y^{k+1}$

(c) $t^{k+1} = \frac{\sqrt{1+4(t^k)^2}+1}{2}$

(d) $y^{k+1} = u^k + \frac{t^k-1}{t^{k+1}}(u^k - u^{k-1})$

until u^k converges or satisfies a stopping criteria.

(перевод с англ.)

Дано: Зашифрованное изображение $x \in \mathbb{R}^d$; $\lambda > 0$, $\beta > 0$, $\gamma > 0$ такие, что $\frac{1}{\gamma\beta} < \frac{1}{4}$.

Инициализация: $y^1 = u^0 = x$, $u^{-1} = 0$, $t^1 = 1$.

Повторять:

1. Обновить компоненты y^{k+1} согласно уравнению (55).
2. $u^k = y^{k+1}$.
3. $t^{k+1} = \frac{\sqrt{1+4(t^k)^2}+1}{2}$.
4. $y^{k+1} = u^k + \frac{t^k-1}{t^{k+1}}(u^k - u^{k-1})$.

Пока u^k сходится или удовлетворяет критерию остановки.

Из другой статьи

1D TV Denoising Algorithm

Input: integer size $N \geq 1$, real sequence $(y[1], \dots, y[N])$, real parameter $\lambda > 0$. Output: real sequence $(x^\star[1], \dots, x^\star[N])$ solution to (1).

1. Set $k = k_0 = k_- = k_+ \leftarrow 1$, $v_{\min} \leftarrow y[1] - \lambda$, $v_{\max} \leftarrow y[1] + \lambda$, $u_{\min} \leftarrow \lambda$, $u_{\max} \leftarrow -\lambda$.
2. If $k = N$, set $x^\star[N] \leftarrow v_{\min} + u_{\min}$ and terminate.
3. If $y[k+1] + u_{\min} < v_{\min} - \lambda$, set $x^\star[k_0] = \dots = x^\star[k_-] \leftarrow v_{\min}$, $k = k_0 = k_- = k_+ \leftarrow k_- + 1$, $v_{\min} \leftarrow y[k]$, $v_{\max} \leftarrow y[k] + 2\lambda$, $u_{\min} \leftarrow \lambda$, $u_{\max} \leftarrow -\lambda$.
4. Else, if $y[k+1] + u_{\max} > v_{\max} + \lambda$, set $x^\star[k_0] = \dots = x^\star[k_+] \leftarrow v_{\max}$, $k = k_0 = k_- = k_+ \leftarrow k_+ + 1$, $v_{\min} \leftarrow y[k] - 2\lambda$, $v_{\max} \leftarrow y[k]$, $u_{\min} \leftarrow \lambda$, $u_{\max} \leftarrow -\lambda$.
5. Else, set $k \leftarrow k+1$, $u_{\min} \leftarrow u_{\min} + y[k] - v_{\min}$ and $u_{\max} \leftarrow u_{\max} + y[k] - v_{\max}$.
6. If $u_{\min} \geq \lambda$, set $v_{\min} \leftarrow v_{\min} + (u_{\min} - \lambda)/(k - k_0 + 1)$, $u_{\min} \leftarrow \lambda$, $k_- \leftarrow k$.
   If $u_{\max} \leq -\lambda$, set $v_{\max} \leftarrow v_{\max} + (u_{\max} + \lambda)/(k - k_0 + 1)$, $u_{\max} \leftarrow -\lambda$, $k_+ \leftarrow k$.
7. If $k < N$, go to 3.
8. If $u_{\min} < 0$, set $x^\star[k_0] = \dots = x^\star[k_-] \leftarrow v_{\min}$, $k = k_0 = k_- \leftarrow k_- + 1$, $v_{\min} \leftarrow y[k]$, $u_{\min} \leftarrow \lambda$, $u_{\max} \leftarrow y[k] + \lambda - v_{\max}$. Then go to 2.
9. Else, if $u_{\max} > 0$, set $x^\star[k_0] = \dots = x^\star[k_+] \leftarrow v_{\max}$, $k = k_0 = k_+ \leftarrow k_+ + 1$, $v_{\max} \leftarrow y[k]$, $u_{\max} \leftarrow -\lambda$, $u_{\min} \leftarrow y[k] - \lambda - v_{\min}$. Then go to 2.
10. Else, set $x^\star[k_0] = \dots = x^\star[N] \leftarrow v_{\min} + u_{\min}/(k - k_0 + 1)$ and terminate.