Chapter 3
Convex optimization for variational image processing

Variational Image Processing
Summer School on Inverse Problems 2015

Michael Moeller
Computer Vision
Department of Computer Science
TU München

Form of the minimization problems

Assumptions

Throughout this chapter, I will assume we have discretized all minimization problems. In other words, our problems are of the form

$$\tilde{u} \in \arg\min_{u \in \mathbb{R}^n} G(u) + F(Ku) \tag{1}$$

for a matrix K and extended real valued functions $G : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$, $F : \mathbb{R}^m \to \mathbb{R} \cup \{\infty\}$, and a matrix $K \in \mathbb{R}^{m \times n}$.

Example problem

Throughout the whole chapter, the classical 1d TV-denoising problem arising from (1) via

$$G(u) = \frac{1}{2}\|u - f\|_2^2 = \frac{1}{2}\sum_i (u_i - f_i)^2$$

$$F(Ku) = \alpha \sum_{i>1} |u_i - u_{i-1}| = \alpha \|Ku\|_1$$

will serve as an example.

Example problem

Throughout the whole chapter, the classical 1d TV-denoising problem arising from (1) via

$$G(u) = \frac{1}{2}\|u - f\|_2^2 = \frac{1}{2}\sum_i (u_i - f_i)^2$$

$$F(Ku) = \alpha \sum_{i>1} |u_i - u_{i-1}| = \alpha \|Ku\|_1$$

will serve as an example.

Completely analog things can be done in 2d.

Domain and properness

Definition

- For $E : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$, we call

$$\text{dom}(E) := \{u \in \mathbb{R}^n \mid E(u) < \infty\}$$

  the domain of E.
- We call E proper if $\text{dom}(E) \neq \emptyset$.

Form of the minimization problems

More assumptions...

In our problems

$$\tilde{u} \in \arg\min_{u \in \mathbb{R}^n} G(u) + F(Ku) \tag{1}$$

we assume

- F and G are convex.

Form of the minimization problems

More assumptions...

In our problems

$$\tilde{u} \in \arg\min_{u \in \mathbb{R}^n} G(u) + F(Ku) \tag{1}$$

we assume

- F and G are convex.
- $\text{dom}(G) \cap \text{dom}(F \circ K) \neq \emptyset$.

Form of the minimization problems

More assumptions...

In our problems

$$\tilde{u} \in \arg\min_{u \in \mathbb{R}^n} G(u) + F(Ku) \tag{1}$$

we assume

- F and G are convex.
- $\text{dom}(G) \cap \text{dom}(F \circ K) \neq \emptyset$.
- F and G are lower-semi continuous, i.e.

  $$\liminf_{v \to u} E(v) \geq E(u)$$

  holds for $E = F$ and $E = G$.

Form of the minimization problems

More assumptions...

In our problems

$$\tilde{u} \in \arg\min_{u \in \mathbb{R}^n} G(u) + F(Ku) \tag{1}$$

we assume

- F and G are convex.
- $\text{dom}(G) \cap \text{dom}(F \circ K) \neq \emptyset$.
- F and G are lower-semi continuous, i.e.

  $$\liminf_{v \to u} E(v) \geq E(u)$$

  holds for $E = F$ and $E = G$.
- $G(u) + F(Ku)$ is *coercive*, i.e.

  $$G(u) + F(Ku) \to \infty \text{ for } \|u\| \to \infty$$

Form of the minimization problems

Existence of minimiziers

Under the above assumptions

$$\tilde{u} \in \arg\min_{u \in \mathbb{R}^n} G(u) + F(Ku)$$

exists.

Summary of the assumptions

- G and F are convex (and not crazy).
- The energy can ”control” $\|u\|$.
- All constraints are $\leq$ or $\geq$ and never $<$ or $>$.

Variational Problems

What is an optimality condition for

$$\hat{u} \in \arg\min_{u \in \mathbb{R}^n} E(u)?$$

Variational Problems

What is an optimality condition for

$$\hat{u} \in \arg\min_{u \in \mathbb{R}^n} E(u)?$$

Definition: Subdifferential

We call

$$\partial E(u) = \{p \in \mathbb{R}^n \mid E(v) - E(u) - \langle p, v - u \rangle \geq 0, \ \forall v \in \mathbb{R}^n\}$$

the subdifferential of E at u.

- Elements of $\partial E(u)$ are called subgradients.
- If $\partial E(u) \neq \emptyset$, we call E subdifferentiable at E.
- By convention, $\partial E(u) = \emptyset$ for $u \neq \mathrm{dom}(E)$.

Variational Problems

What is an optimality condition for

$$\hat{u} \in \arg\min_{u \in \mathbb{R}^n} E(u)?$$

Definition: Subdifferential

We call

$$\partial E(u) = \{p \in \mathbb{R}^n \mid E(v) - E(u) - \langle p, v - u \rangle \geq 0, \ \forall v \in \mathbb{R}^n\}$$

the subdifferential of E at u.

- Elements of $\partial E(u)$ are called subgradients.
- If $\partial E(u) \neq \emptyset$, we call E subdifferentiable at E.
- By convention, $\partial E(u) = \emptyset$ for $u \neq \text{dom}(E)$.

Theorem: Optimality condition

Let $0 \in \partial E(\hat{u})$. Then $\hat{u} \in \arg\min_u E(u)$.

Variational Problems

Examples for non-differentiable functions:

- The ℓ^1 norm.

Variational Problems

Examples for non-differentiable functions:

- The ℓ^1 norm.
- Functional

$$E(u) = \begin{cases} 0 & \text{if } u \in [0, \infty[\\ \infty & \text{else.} \end{cases}$$

Variational Problems

Examples for non-differentiable functions:

- The ℓ^1 norm.
- Functional

$$E(u) = \begin{cases} 0 & \text{if } u \in [0, \infty[\\ \infty & \text{else.} \end{cases}$$

Definition: Relative Interior

The *relative interior* of a convex set M is defined as

$$\text{ri}(M) := \{x \in M \mid \forall y \in M, \ \exists \lambda > 1, \ \text{s.t. } \lambda x + (1 - \lambda) y \in M\}$$

Theorem: Subdifferentiability[a]

[a] Rockafellar, Convex Analysis, Theorem 23.4

If E is a proper convex function and $u \in \text{ri}(\text{dom}(E))$, then $\partial E(u)$ is non-empty and bounded.

Variational Problems

Subdifferential calculus

- **Generalized derivative:** If E is differentiable at u, then

$$\partial E(u) = \{\nabla E(u)\}.$$

Variational Problems

Subdifferential calculus

- **Generalized derivative:** If E is differentiable at u, then

$$\partial E(u) = \{\nabla E(u)\}.$$

- **Sum rule:** If $\text{ri}(\text{dom}(E_1)) \cap \text{ri}(\text{dom}(E_2)) \neq \emptyset$, then

$$\partial(E_1 + E_2)(u) = \partial E_1(u) + \partial E_2(u)$$

Variational Problems

Subdifferential calculus

- **Generalized derivative:** If E is differentiable at u, then

$$\partial E(u) = \{\nabla E(u)\}.$$

- **Sum rule:** If $\text{ri}(\text{dom}(E_1)) \cap \text{ri}(\text{dom}(E_2)) \neq \emptyset$, then

$$\partial(E_1 + E_2)(u) = \partial E_1(u) + \partial E_2(u)$$

- **Cain rule:** If $\text{ri}(\text{dom}(E)) \cap \text{range}(A) \neq \emptyset$ then

$$\partial(E \circ A)(u) = A^* \partial E(Au)$$

Example problems with explicit solution

Examples

$$\hat{u} = \min_u \frac{1}{2}\|u - v\|_2^2 + \alpha E(u) \qquad \text{(prox)}$$

for

- $E(u) = \frac{1}{2}\|u - f\|_2^2$
- $E(u) = \|u\|_1$

Example problems with explicit solution

Examples

$$\hat{u} = \min_u \frac{1}{2}\|u - v\|_2^2 + \alpha E(u) \qquad \text{(prox)}$$

for

- $E(u) = \frac{1}{2}\|u - f\|_2^2$
- $E(u) = \|u\|_1$

Observation: For typical choices of F and G, problem (prox) (with $E = F$ or $E = G$) is easy to solve.

Relation to convex conjugate

Fenchel-Young Inequality[a]

[a] Borwein, Zhu *Techniques of variational analysis*, Proposition 4.4.1

Let E be proper, convex and lower semi-continuous, $u \in \text{dom}(E) \subset \mathbb{R}^n$, and $p \in \mathbb{R}^n$, then

$$E(u) + E^*(p) \geq \langle u, p \rangle.$$

Equality holds if and only if $p \in \partial E(u)$.

Relation to convex conjugate

Fenchel-Young Inequality[a]

[a]Borwein, Zhu *Techniques of variational analysis*, Proposition 4.4.1

Let E be proper, convex and lower semi-continuous, $u \in \text{dom}(E) \subset \mathbb{R}^n$, and $p \in \mathbb{R}^n$, then

$$E(u) + E^*(p) \geq \langle u, p \rangle.$$

Equality holds if and only if $p \in \partial E(u)$.

Theorem: Subgradient of convex conjugate[a]

[a]Rockafellar, Convex Analysis, Theorem 23.5

Let E be proper, convex and lower semi-continuous, then the following two conditions are equivalent:

- $p \in \partial E(u)$
- $u \in \partial E^*(p)$

Optimality condition

$$\tilde{u} \in \arg\min_{u \in \mathbb{R}^n} G(u) + F(Ku)$$

Optimality condition

$$\tilde{u} \in \arg\min_{u \in \mathbb{R}^n} G(u) + F(Ku)$$

Optimality condition

$$0 \in \partial G(u) + K^* \partial F(Ku).$$

Optimality condition

$$\tilde{u} \in \arg\min_{u \in \mathbb{R}^n} G(u) + F(Ku)$$

Optimality condition

$$0 \in \partial G(u) + K^* \partial F(Ku).$$

Introduce $q \in \partial F(Ku)$ and use convex conjugate to obtain

$$\begin{aligned} 0 &\in \partial F^*(\tilde{q}) - K\tilde{u} \\ 0 &\in \partial G(\tilde{u}) \;\; + K^* \tilde{q} \end{aligned}$$

or in stacked form:

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} \tilde{q} \\ \tilde{u} \end{pmatrix}$$

Optimality condition

We need to find $(\tilde{q}, \tilde{u})$ with

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} \tilde{q} \\ \tilde{u} \end{pmatrix},$$

but how?

The proximal point algorithm

Monotone operators

A set valued operator T is called monotone, if for all z_1, z_2, and $p_1 \in Tz_1$, $p_2 \in Tz_2$ it holds that

$$\langle p_1 - p_2, z_1 - z_2 \rangle \geq 0.$$

The proximal point algorithm

Monotone operators

A set valued operator T is called monotone, if for all z_1, z_2, and $p_1 \in Tz_1$, $p_2 \in Tz_2$ it holds that

$$\langle p_1 - p_2, z_1 - z_2 \rangle \geq 0.$$

Proximal point algorithm (PPA)

Good candidate for finding a point $\hat{z}$ with $0 \in T\hat{z}$ for T being monotone:

$$0 \in Tz^{k+1} + M(z^{k+1} - z^k)$$

for a symmetric positive definite matrix M.

Rockafellar, *Monotone operators and the proximal point algorithm*.
Eckstein, *Splitting methods for monotone operators with applications to parallel optimization*.

The proximal point algorithm

Applicability of PPA

The operator

$$T = \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix}$$

is monotone.

The proximal point algorithm

Applicability of PPA

The operator

$$T = \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix}$$

is monotone.

Conclusion: The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & M_2 \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

will converge to a solution of our minimization problem, if M is positive definite.

The proximal point algorithm

Applicability of PPA

The operator

$$T = \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix}$$

is monotone.

Conclusion: The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & M_2 \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

will converge to a solution of our minimization problem, if M is positive definite.

We need to ensure two things with the M_i:

1. Make sure each iteration is easy to evaluate.
2. Make sure M is symmetric positive definite.

The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & M_2 \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Step 1: Make sure each iteration is easy to evaluate.

The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & M_2 \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Step 1: Make sure each iteration is easy to evaluate.
Choose $M_2 = K$!

The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & M_2 \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Step 1: Make sure each iteration is easy to evaluate.
Choose $M_2 = K$!

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & K \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Advantage: The first inclusion is independent of u^{k+1}! Given u^k and q^k, solve for q^{k+1}!

Remark: An alternate choice would have been $M_3 = -K^*$ with a similar effect for u^{k+1}.

The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & K \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Step 2: Make sure M symmetric positive definite.

The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & K \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Step 2: Make sure M symmetric positive definite.
A simple option is $M_1 = \frac{1}{\tau} I$, $M_4 = \frac{1}{\sigma}$, $M_3 = K^*$, $\tau\sigma \|K\|_2^2 < 1$.

Michael Moeller

The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & K \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Step 2: Make sure M symmetric positive definite.
A simple option is $M_1 = \frac{1}{\tau} I$, $M_4 = \frac{1}{\sigma}$, $M_3 = K^*$, $\tau\sigma \|K\|_2^2 < 1$.

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} \frac{1}{\tau} I & K \\ K^* & \frac{1}{\sigma} I \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & K \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Step 2: Make sure M symmetric positive definite.
A simple option is $M_1 = \frac{1}{\tau} I$, $M_4 = \frac{1}{\sigma}$, $M_3 = K^*$, $\tau\sigma\|K\|_2^2 < 1$.

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix}}_{=:T} \underbrace{\begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix}}_{=:z^{k+1}} + \underbrace{\begin{pmatrix} \frac{1}{\tau} I & K \\ K^* & \frac{1}{\sigma} I \end{pmatrix}}_{=:M} \underbrace{\begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix}}_{z^{k+1} - z^k},$$

The proximal point algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & K \\ M_3 & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Step 2: Make sure M symmetric positive definite.
A simple option is $M_1 = \frac{1}{\tau} I$, $M_4 = \frac{1}{\sigma}$, $M_3 = K^*$, $\tau\sigma\|K\|_2^2 < 1$.

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \underbrace{\begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix}}_{=:T} \underbrace{\begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix}}_{=:z^{k+1}} + \underbrace{\begin{pmatrix} \frac{1}{\tau} I & K \\ K^* & \frac{1}{\sigma} I \end{pmatrix}}_{=:M} \underbrace{\begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix}}_{z^{k+1} - z^k},$$

$$0 \in T z^{k+1} + M(z^{k+1} - z^k)$$

– Proximal point algorithm –

Computing the updates

And how do we actually do the updates?

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} \frac{1}{\tau} I & K \\ K^* & \frac{1}{\sigma} I \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Computing the updates

And how do we actually do the updates?

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} \frac{1}{\tau} I & K \\ K^* & \frac{1}{\sigma} I \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Update for q:

$$0 \in \partial F^*(q^{k+1}) + \frac{1}{\tau}(q^{k+1} - q^k - \tau K u^k)$$

$$q^{k+1} = \arg\min_q \left(\frac{1}{2\tau} \|q - q^k - \tau K u^k\|_2^2 + F^*(q) \right)$$

Computing the updates

And how do we actually do the updates?

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} \frac{1}{\tau} I & K \\ K^* & \frac{1}{\sigma} I \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Update for q:

$$0 \in \partial F^*(q^{k+1}) + \frac{1}{\tau}(q^{k+1} - q^k - \tau K u^k)$$

$$q^{k+1} = \arg\min_q \left(\frac{1}{2\tau} \|q - q^k - \tau K u^k\|_2^2 + F^*(q) \right)$$

For $F^*(q) = \mathrm{i}_{\|\cdot\|_\infty \leq \alpha}(q)$ one obtains

$$q_i^{k+1} = \begin{cases} (q^k + \tau K u^k)_i & \text{if } (q^k + \tau K u^k)_i \in [-\alpha, \alpha] \\ \alpha & \text{if } (q^k + \tau K u^k)_i > \alpha \\ -\alpha & \text{if } (q^k + \tau K u^k)_i < -\alpha \end{cases}$$

Computing the updates

And how do we actually do the updates?

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} \frac{1}{\tau} I & K \\ K^* & \frac{1}{\sigma} I \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Update for u:

$$0 \in \partial G(u^{k+1}) + \frac{1}{\sigma} \left(u^{k+1} - u^k + \sigma K^*(2q^{k+1} - q^k) \right)$$

$$u^{k+1} = \arg\min_q \left(\frac{1}{2\sigma} \|u - u^k + \sigma K^*(2q^{k+1} - q^k)\|_2^2 + G(u) \right)$$

Computing the updates

And how do we actually do the updates?

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} \frac{1}{\tau} I & K \\ K^* & \frac{1}{\sigma} I \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

Update for u:

$$0 \in \partial G(u^{k+1}) + \frac{1}{\sigma} \left(u^{k+1} - u^k + \sigma K^*(2q^{k+1} - q^k) \right)$$

$$u^{k+1} = \arg\min_q \left(\frac{1}{2\sigma} \|u - u^k + \sigma K^*(2q^{k+1} - q^k)\|_2^2 + G(u) \right)$$

For $G(u) = \frac{1}{2}\|u - f\|_2^2$ one obtains

$$u^{k+1} = \frac{1}{1+\sigma} \left(u^k + \sigma f - \sigma K^*(2q^{k+1} - q^k) \right)$$

1d TV minimization

Primal-dual TV-minimization algorithm:

$$q_i^{k+1} = \begin{cases} (q^k + \tau K u^k)_i & \text{if } (q^k + \tau K u^k)_i \in [-\alpha, \alpha] \\ \alpha & \text{if } (q^k + \tau K u^k)_i > \alpha \\ -\alpha & \text{if } (q^k + \tau K u^k)_i < -\alpha \end{cases}$$

$$u^{k+1} = \frac{1}{1+\sigma}\left(u^k + \sigma f - \sigma K^*(2q^{k+1} - q^k)\right)$$

The general algorithm

Solving

$$\min_u G(u) + F(Ku)$$

Primal-dual minimization algorithm:

$$q^{k+1} = \arg\min_q \left(\frac{1}{2\tau} \|q - q^k - \tau K u^k\|_2^2 + F^*(q) \right)$$

$$u^{k+1} = \arg\min_q \left(\frac{1}{2\sigma} \|u - u^k + \sigma K^*(2q^{k+1} - q^k)\|_2^2 + G(u) \right)$$

with $F^*(q) = \sup_v \langle q, v\rangle - F(v)$ generalizing $\mathfrak{i}_{\|\cdot\|_\infty \leq \alpha}(q)$.

A Convex Relaxation Approach for Computing Minimal Partitions, Chambolle, Cremers, Bischof, Pock.
An Efficient Primal-Dual Hybrid Gradient Algorithm for Total Variation Image Restoration, Zhu, Chan.
A General Framework for a Class of First Order Primal-Dual Algorithms for Convex Optimization in Imaging Science. Esser, Chan, Zhang.
A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging, Chambolle, Pock.

Deriving other methods

Framework

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial F^* & -K \\ K^* & \partial G \end{pmatrix} \begin{pmatrix} q^{k+1} \\ u^{k+1} \end{pmatrix} + \begin{pmatrix} M_1 & K \\ K^* & M_4 \end{pmatrix} \begin{pmatrix} q^{k+1} - q^k \\ u^{k+1} - u^k \end{pmatrix},$$

allows to derive many more interesting algorithms.

- $M_1 = \lambda K K^*$, $M_4 = \frac{1}{\lambda} I$: ADMM
- $M_1 = \lambda I$, $M_4 = \frac{1}{\lambda} K^* K$: ADMM
- Diagonal matrices, e.g. in the framework of
  - Zhang, Burger, Osher, *A unified primal-dual algorithm framework based on Bregman iteration.*
  - Pock, Chambolle, *Diagonal preconditioning for first order primal-dual algorithms in convex optimization*.

Further overviews and extensions:

- *First Order Algorithms in Variational Image Processing*. Burger, Sawatzky, Steidl.
- *An inertial forward-backward method for monotone inclusions*. Lorenz, Pock.

Summary

Interested in trying things out yourself?

Matlab implementation + GUI for solving Denoising, Deblurring, Zooming, Inpainting, Motion Estimation, and Segmentation with the presented primal-dual algorithm:
`http://gpu4vision.icg.tugraz.at/index.php?content=downloads.php` [1]

[1] Replace *isrgb(img)* with *size(img,3)==3*

Summary

Interested in trying things out yourself?

Matlab implementation + GUI for solving Denoising, Deblurring, Zooming, Inpainting, Motion Estimation, and Segmentation with the presented primal-dual algorithm:
`http://gpu4vision.icg.tugraz.at/index.php?content=downloads.php` [1]

Open positions: `http://vision.in.tum.de/jobs`

[1] Replace *isrgb(img)* with *size(img,3)==3*

Summary

Interested in trying things out yourself?

Matlab implementation + GUI for solving Denoising, Deblurring, Zooming, Inpainting, Motion Estimation, and Segmentation with the presented primal-dual algorithm:
`http://gpu4vision.icg.tugraz.at/index.php?content=downloads.php` [1]

Open positions: `http://vision.in.tum.de/jobs`

Thank you!

michael.moeller@in.tum.de

[1] Replace *isrgb(img)* with *size(img,3)==3*