Motion correction for medical imaging

Camille Pouchol

CIAM, KTH Royal Institute of Technology, Stockholm

***The mathematics of imaging* winter school,**
CIRM, Marseille, 7-11 January 2019

Problem set up

Assume we know how to "invert" some linear ill-posed inverse problem $Af_0 = g_0$ (actually, g_0 is a Poisson or Gaussian with mean Af_0), by solving something like

$$\arg\min_{f_0} \; d(Af_0, g_0) \, (+R(f_0)).$$

In Positron Emission Tomography (PET), A is the so-called *system matrix*, g_0 is the data we have, f_0 is the image we are looking for.

Problem set up

Assume we know how to "invert" some linear ill-posed inverse problem $Af_0 = g_0$ (actually, g_0 is a Poisson or Gaussian with mean Af_0), by solving something like

$$\arg\min_{f_0} \; d(Af_0, g_0)\,(+R(f_0)).$$

In Positron Emission Tomography (PET), A is the so-called *system matrix*, g_0 is the data we have, f_0 is the image we are looking for.

Problem: what if the organ is moving? (lungs, heart)

The problem becomes $Af(t) = g(t)$, and we want to recover, say, $f(0)$.

Diffeomorphisms

In practice, possibility to group phases together. Example: PET imaging for lungs, 15 to 20 minutes.

For simplicity, assuming two gates (like inhale and exhale positions):

Two images f_0 and f_1, and you observe g_0 and g_1, (a stochastic version of) Af_0 and Af_1.

Goal: recover f_0.

Final step of modelling: f_1 and f_0 are linked by a transformation, some **unknown** diffeomorphism φ such that $f_1 = \varphi \circ f_0$.

In practice, possibility to group phases together. Example: PET imaging for lungs, 15 to 20 minutes.

For simplicity, assuming two gates (like inhale and exhale positions):

Two images f_0 and f_1, and you observe g_0 and g_1, (a stochastic version of) Af_0 and Af_1.

Goal: recover f_0.

Final step of modelling: f_1 and f_0 are linked by a transformation, some **unknown** diffeomorphism φ such that $f_1 = \varphi \circ f_0$.

$$\arg\min_{f_{\mathbf{0}},\varphi} \; d(Af_0, g_0) + d(Af_1, g_1)\,(+R(f_0) + R(f_1) + R(\varphi)).$$

Difficulty: requires recovering both f_0, this is *reconstruction* AND φ, this is *motion correction*.

Intertwining

Idea: solve the optimisation problem by alternatively trying to find f_0 (reconstruction), then φ (motion estimation), each phase improving each other. By gradient descent?

Problems:

- unfeasible in practice, matrix A is huge and only a few iterations are allowed,
- if only a few iterations, terrible results,
- searching among diffeomorphisms can also be terribly cumbersome.

Intertwining

Idea: solve the optimisation problem by alternatively trying to find f_0 (reconstruction), then φ (motion estimation), each phase improving each other. By gradient descent?

Problems:

- unfeasible in practice, matrix A is huge and only a few iterations are allowed,
- if only a few iterations, terrible results,
- searching among diffeomorphisms can also be terribly cumbersome.

We are trying

- (reconstruction) to use expectation maximisation (ML-EM) so that even a few iterations yield coherent results,
- (motion correction) to learn the diffeomorphisms: direct or indirect matching by parametrising with neural networks.