Group-wise Non-rigid Strategy and Notation

The following notes aim to define some terminology and symbolic representation of registration methodology. We seek a *symbolic* definition of algorithms rather than verbal ones.

Structure of the Notes

1. Notation for registration algorithms
2. Graphical Representation
3. Use (1) and (2) to define current approaches
4. Proposal of new approaches

What elements does registration involve?

- Warps
- Similarity Measures
- Discrepancy/Residual error
- Models (e.g. shape, combined)
- Images (or volumes, or vectors)

Examples of connection between these elements

- Warps are applied to images
- Discrepancy guides choice of warps
- Models are representative of the images
- Similarity measures are derived from images

Types of connectivity

- Inference
- Backward inference
- Collective inference
- Transformation
- Optimisation
- Data flow

Which is which?

Quite clearly, there is great overlap between the categories mentioned. However, there is a relatively small number of connection types and it is sensible to formally define a notion to each.

Connectivity type #1: Warps

These are relatively simple. We have a warp donated by w and it is defined by some parameters e.g. knot-points. This gives a more general form of $w(data, p_1, p_2, p_3, ..., p_n)$.

The inverse is $w'(data, p_1, p_2, p_3, ..., p_n)$.

Ideally, $\forall x \; w'(w(x, p_1, p_2, p_3, ..., p_n), p_1, p_2, p_3, ..., p_n)) = x$

Graphically, let us use a double-lined arrow $\Rightarrow$ to indicate actual *change* to data made.

Connectivity type #2: Inference

Best explained using examples

- An image implies its given histogram.
- A model implies its assigned complexity

Can be simply represented by a standard arrow $\rightarrow$.

Connectivity type #3: Backward Inference

This is essentially an information pull-back – derive something from given data. This is an inference where an object *requests* data from another and learns from this data. Algorithmically, data flows backwards upon signal. A broken arrow $\rightharpoonup$ can be used to stress that the invocation of data transfer is really initiated by the receiver although it is sent in the direction of the arrow.

Connectivity type #4: Collective Inference

- Several elements are used to derive one in this case. Forking an arrow should be trivial. That notation will often be used to visualise model construction.

What else?

- We need to denote optimisation over a collection of mixed parameters. This will help us to keep track of properties over which we optimise, i.e. the components of the objective function. We could use **bold** style to indicate which parameters these are.

- Data flow was implicitly included in cases (2)-(4). However, if no inference is involved, a line with no arrow can be used. This makes it difficult to understand in which direction data goes. Concise annotation as in UML can then be added.

...Continued

- Notation for commonly used functions needs to be agreed upon e.g. **hist()** and **msd()**. These can be used alongside arrows.

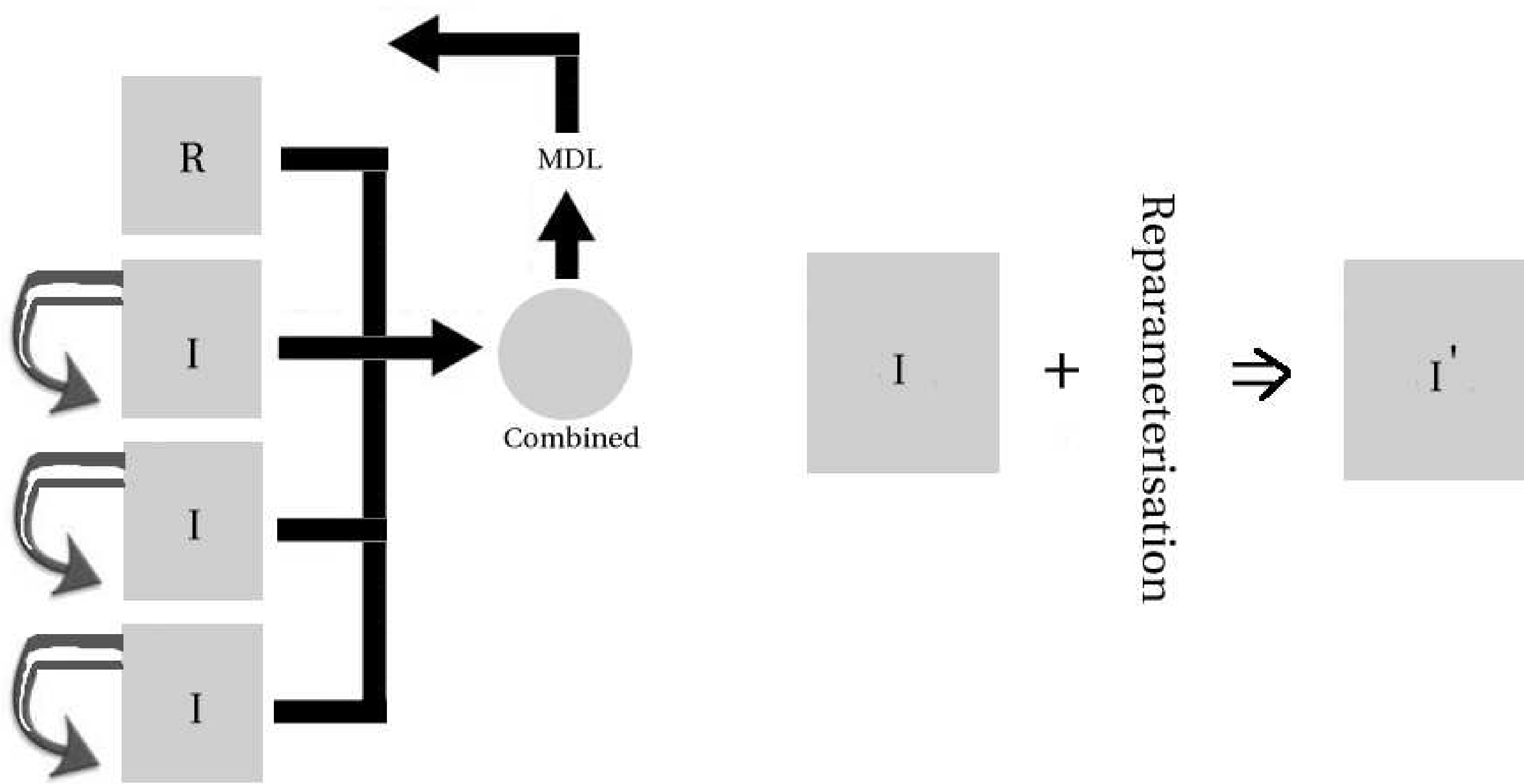
- Visual representation of components in the system needs to be consistent. The next side proposed attempts to affix some standard shapes to entities.

Entities Representation

- Let us use a circle to represent a model with its type description in the centre.

- Data is presumed to be images so a rectangle should do. R denotes "reference", T denotes "target", and I is set to be the default type "image" which can be omitted.

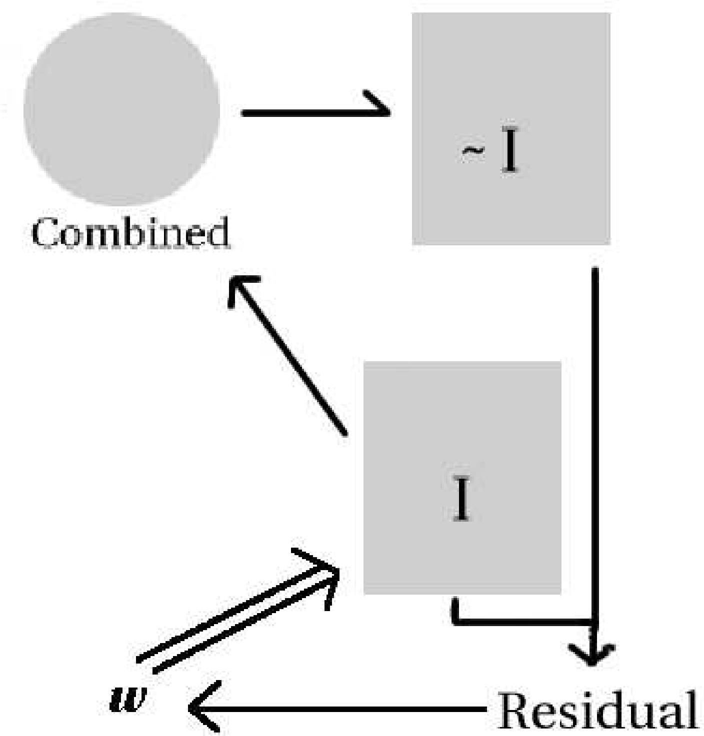
Current Method Graphically



Current Method in a Nutshell

- A reference immutable **R** is at the top.
- For each image **I**, a warp w transforms it to result in a combined model with certain properties.
- $log \prod \lambda$ is computed to infer MDL of the combined model
- MDL guides optimisation over w.
- Warps are established using a reparameterisation followed by curve interpolation.

Proposed Method Graphically



Proposed Methods Explained

1. A reconstruction is inferred from the model using the parameters b_i.

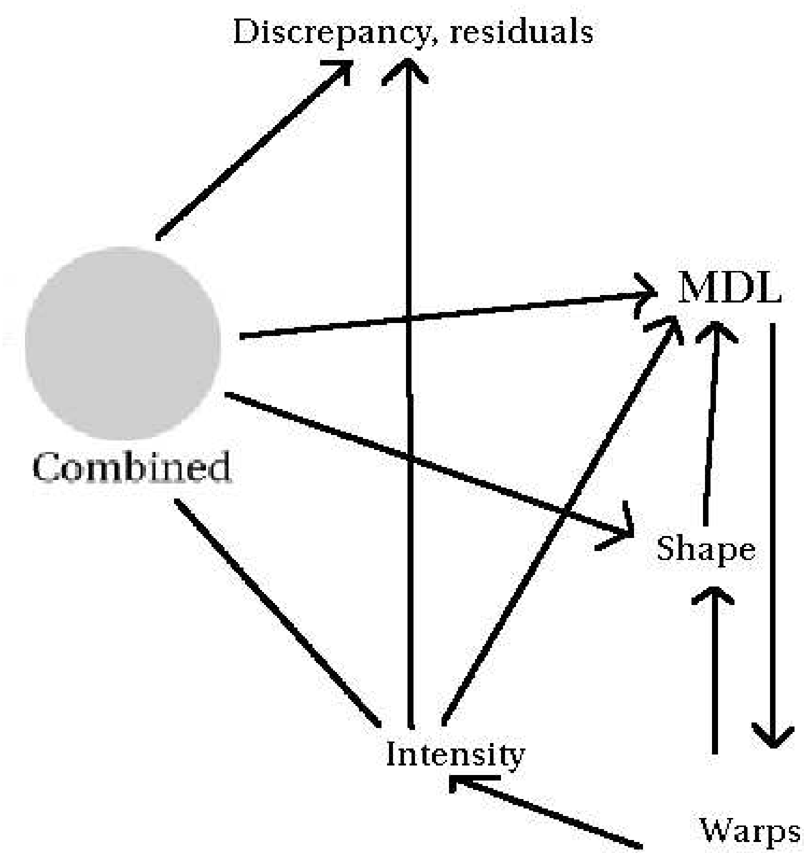
2. The reconstruction is compared to the original data.

3. A residual is the product of comparing (1) and (2).

4. The residual guides a set of new warps, to be determined by the optimiser.

5. Warps are applied to the data which then updates the model.

Proposed Method: Internal Intricacies



Proposed Methods: Further Explanation

Let us look at some of the relationships in the system...

- There is a great number of dependencies
- The observations made show possible integrations in the system.
- Each component has some possible relation (forward, backwards, or transitive) to another.
- The correct 'recipe' for registration and warp modelling may be out there.

Proposed Methods Continued

- The relationships must be well-understood first.

- Questions yet to be answered:

  1. How can we prevent the data from drifting away?
  2. How should residuals be involved in the objective function?
  3. Is the determinant a sufficiently good measure of description length?

  There must be more...

Summary

- We identified the elements in the system
- We defined some notation for elements in the system
- Current approach was described w.r.t. to the notation
- New methods were proposed and explained.
- Potentially better objective function will soon be investigated.