Devising an explicit algorithm based on simple rules is difficult! L1 reg: $\ell = err(y, \hat{y}) + \lambda \sum_{i=1}^{N} |w_i|$ favours few non-0 coefs, L2 favours small coefs
under-fitting → high bias (high training, high test error) → add features, decrease regularization term λ, increase degree of polynomial)
over-fitting → high variance (low training, high test error) → get more data, remove features, increase regularization term λ, decrease degree of polynom)

Linear Transformations

$$\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = \underbrace{\overbrace{\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}}^{\text{translation vectors}} \overbrace{\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}^{\text{direction vectors}} \overbrace{\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}^{\text{scaling matrix}}}_{T_{ItW}} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

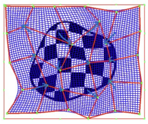
If there is more than one world, then:

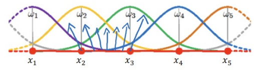
$$\begin{pmatrix} x_A \\ y_A \end{pmatrix} = \mathrm{T}^A_{WtI} T_{BtA} T^B_{ItW} \begin{pmatrix} x_B \\ y_B \end{pmatrix}$$

In 3D, there are identity (0 dof) rigid (translation and rotation, 3+3 dof), similar (scaling, 3+3+1 dof), affine (shear 12). In affine, if two lines are ||, after affine T, its still true.

$$\text{shearing} = \begin{bmatrix} \cos\omega & \sin\omega & 0 \\ -\sin\omega & \cos\omega & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Non-linear Transformations





Embed image onto grid, prescribe motion at each grid point (red) and underlying blue grid is as a result of linear interpolation.
Medical Applications: Cardiac Motion and Respiration Motion tracking, Multi-modal Image Fusion, Pre- and Post-op comparison
Intra-Subject Registration: create an atlas and transform.

Challenges in Semantic Segmentation (every pixel in an image belongs to a class)

- *noise* – high-frequency pixel variability (not relevant/may obscure target)
- *partial volume* – quantized version of object (pixels may contain mix of two objects and both contribute to pixel value) and object may be elevated (unclear where to begin/end object)
- *intensity inhomogeneities* – varying contrast and intensity differences across the image plain
- *anisotropic resolution* – (not isotropic, where voxels are cubes) causes ↓ clarity in coarse dims
- *imaging artifacts* – implants may interfere with imaging modality
- *limited contrasts* – different tissues may have similar physical properties and leak boundaries
- *morphological variability* – variability in physiological conditions or imaging modalities

Pitfalls in Segmentation Evaluation

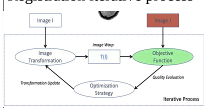
- *Structure Size* – equal differences between small and big structures change spatial overlap lots
- *Structure Shape* – spatial overlap metrics are unaware of complex shapes
- *Spatial alignment* – HD & DSC & IoU don't capture object centre point alignment
- *Holes* – Boundary-based metrics ignore overlap between structures
- *Noise* – Affects HD as it is spiked by a far away FP
- *Empty Label-Maps* – scores of 0 or NaN for each method with combo of empty ref. or predict.
- *Resolution* – same prediction shapes at different resolutions give different results
- *Over vs Under-segmentation* – for equal HD, DSC may be better for over than undersegment.

Segmentation Methods

- *Intensity Segmentation:* (hist) × regions must be homogenous, leakages, threshold loc. Hard
- *Region Based*: (start from seed) × requires user points, leakages, assumed homogeneity
- *Atlas Based:* (averaged templates) *Registration:* mutate multiple atlases into target and fuse labels (majority voting) (this saves pdf indicating contention between sources). √ robust, accurate, automatic × comput, expensive, poor for abnormalities, not for tumour segmentation.
- *Random Forests:* different modalities of 1 image, construct a tree to classify a pixel based on rules. Ensemble it by averaging answer of many trees. × no hierarchal features √ || & accurate

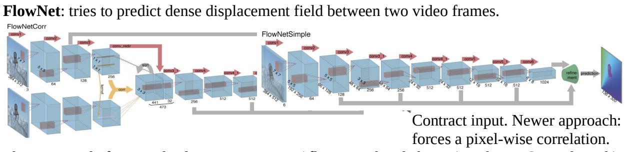
- (up)pooling and max (un)pooling with stored spatial location
- Transposed Convolution:
  $(M-1) \times S - 2P + D \times (K-1) + P_{output} + 1$
- Convolution: $\left\lfloor \frac{M+2P-D*(K-1)-1}{S} \right\rfloor + 1$
- Atrous spatial pyramid pooling: repeated max-pooling and striding reduces spatial resolution of the resulting feature map
- Padding during upsampling may introduce artifacts

Registration iterative process



Gradient descent or SGD.

FlowNet: tries to predict dense displacement field between two video frames.



Contract input. Newer approach: forces a pixel-wise correlation. Then upsample for p.w. displacement. Train w/ flying rendered chairs (you know Ground Truth). Next evolution of **FlowNet2.0** passes image through once, applies the translation, and passes it into the next layer for further fine-tuning. In parallel, there is detailed matching, then concat all.
Optical Flow with Semantic Segmentation and Localized Layers: segment 'Things', 'Planes' & 'Stuff'. Then perform flow estimation on segmented objects for a sharper answer.
Non-rigid Image Registration Using Multi-scale 3D CNNs: randomly deform an image, then train a model to predict your known deformation. To use this network, you need to slide this network across the image and generate for each pixel a displacement vector.
Spatial Transformer Networks: takes feature map (original or pre-processed) and predicts transformation and transform the image according to this transform map. There is a localisation net which trains θ to then deform the grid.
Unsupervised Deformable Image Registration: Two images are fed into an NN to predict deformation. Then feed into spatial transformer, this transforms input and calculates sim. metric.
Voxel Morph: u-net architecture which produces a dense displacement field. Then it uses the spacial transformer to warp the image to the fixed image then minimise the loss to the network.

Generative-based learning: teaching the network to reconstruct an image from a corrupted version.

- Vision Transformers: as above, non-overlapping patches learn representation for patches.
- Masked Auto-encoders are scalable vision learners: divide the image in multiple sub-patches and randomly mask some of the patches. Feed it through the vision transformer. The encoder gets a representation f.e. patch and then fills in the blanks with a mask token you learn during training. The decoder predicts each patch (this is smaller since encoder does most of the work) Loss: $MSE = \sum(\hat{x}_i - x_i)^2, \quad \text{where } i \text{ is the pixel index}$. Masking ratio of >75% is best.

Joint-Embedding Prediction (i-Jepa): a mix of both.
Here, we pass available patches onto the encoder. In ||, sample several other regions and pass it through the same encoder. The task here, is to predict the encoded representation (instead of the output image) instead of wasting compute reconstructing ever single pixel in the image, we are going to reconstruct the important information which there should be if the network is trained summarised. Loss: $\frac{1}{M}\sum_{i=1}^{N} D(\hat{s}_y(i), s_y(i)) = \frac{1}{M}\sum_{i=1}^{M}\sum_{j \in B_i} ||\hat{s}_{y_j} - s_{y_j}||_2^2$.

Output size = , # of param $C \times K \times K$

Expert Gold Standard: × training, tedious, intra (same dude) + inter (diff dude) observability variability, √ multiple segmentations, agreement can be quantified

- specificity = $\frac{TN}{N} = \frac{TN}{TN+FP}$
- Fβ = $(1+\beta^2)\frac{Precision \cdot Recall}{(\beta^2 \cdot Precision)+Recall}$
- Jaccard Index/ IoU = $\frac{|S_g \cap S_p|}{|S_g \cup S_p|} = \frac{DSC}{2-DSC}$
- Dice Sim. Coeff. = $2\frac{|S_g \cap S_p|}{|S_g|+|S_p|} = F_1$
- Volume Sim = $1 - \frac{||S_g|-|S_p||}{|S_g|+|S_p|} = 1 - \frac{|FN-FP|}{2TP+FP+FFN}$

surface distance measure

- Hausdorff Distance = $\max(h(A,B), h(B,A)), h(A,B) = \max_{a \in A}\min_{b \in B}||a-b||$
- Average Surface Distance (create map and swap) $\frac{d(A,B)+d(B,A)}{2}, d(A,B) = \frac{1}{N}\sum_{a \in A}\min_{b \in B}||a-b||$

Multi-scale processing: 4 layers of 5^3 kernels followed by 1^3 kernel for classification. Multiple pathways for different sized snippets of the image. Then we concat. Feature maps from both pathways
Vision transformers: split image into patches, encode location, get hidden feature after convolutions, linear layer and pass through attention network similar to nlp. Upsample in U-net fashion with connections.

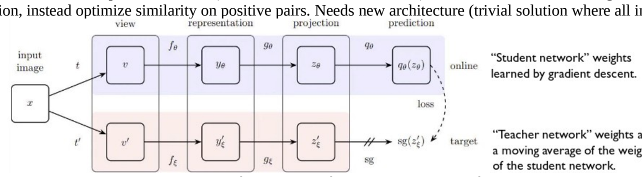
Objective: $C(T) = D(I \circ T, J)$ (**T**ransformation, **D**issimilarity measure, (**J**) Fixed image, $(I \circ T)$ Moving Image
Optimization: $\hat{T} = \arg\min_T C(T)$
Mono-modal Registration: Image intensities are related by a (simple) function. Assumption: the identity relationship between intensity distributions. Not good when brightness changes and subtraction is no longer the best metric.

- Sum of squared differences: $D_{SSD}(I \circ T, J) = \frac{1}{N}\sum_{i=1}^{N}(I(T(x_i)) - J(x_i))^2$
- Sum of absolute differences: $D_{SAD}(I \circ T, J) = \frac{1}{N}\sum_{i=1}^{N}|I(T(x_i)) - J(x_i)|$
- Correlation Coefficient: $D_{CC}(I \circ T, J) = -\frac{\overbrace{\frac{1}{N}\sum_{i=1}^{N}(I(T(x_i)) - \mu_I)(J(x_i) - \mu_I)}^{cov(I,J)}}{\underbrace{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(I(T(x_i)) - \mu_I)^2}}_{\sigma_I}\underbrace{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(J(x_i) - \mu_J)^2}}_{\sigma_I}}$

Multi-modal Registration: Image intensities are related by a complex function or statistical relationship. Measures "When are these two images the most statistically aligned". To avoid local minimas: increase dof, or gaussian smoothing. May require linear interpolation

- Intensity Histograms: plot intensities of both images on x and y, discretize histogram with bins. A Registered image will have more clustered regions (identity will be line $x = y$)
  - $p(i,j) = \frac{h(i,j)}{N}$ at a point, in one image i and other image j counts in the histogram. $p(i) = \sum_j p(i,j)$ sim for $p(j)$
  - Shannon Entropy: $H(I) = -\sum_i p(i)\log p(i)$ low value if every pixel has the same value, or high if randomness
  - Joint Entropy: $H(I,J) = -\sum_i\sum_j p(i,j)\log p(i,j)$ measures how clustered a space is, and minimising that entropy is a good criterium
  - Mutual Information: $MI(I,J) = H(I) + H(J) - H(I,J)$ describes how well one image can be explained by another image. $MI(I,J) = \sum_i\sum_j p(i,j)\log\frac{p(i,j)}{p(i)p(j)}$ with dissimilarity: $D_{MI}(I \circ T, J) = -MI(I \circ T, J)$
  - Normalized Mutual Information: $NMI(I,J) = \frac{H(I)+H(J)}{H(I,J)}$ with dissimilarity similar to above

Contrastive-based learning (SIMCLR): teaching the network to recognise meaningful pairs of images. Idea: create multiple versions of the same image and teach the network to recognise the correct pair. An encoder outputs hidden state h and the small MLP gets a smaller representation (in high dimensional space, comparing similarities is harder, and not necessarily robust). Augmentation Pipeline: Needs to reflect what information the model should disregard and what it should focus on. Needs to be difficult, otherwise, trivial features may be learnt. Normalised temperature contrastive loss: $sim(\vec{u},\vec{v}) = \frac{\vec{u}^\top\vec{v}}{||\vec{u}|| \cdot ||\vec{v}||}$ with Loss $\ell_{i,j} = -\log\frac{\exp(sim(\vec{z}_i,\vec{z}_j)/\tau)}{\sum_{k=1}^{2N}\mathbb{1}_{[k \neq i]}\exp(sim(\vec{z}_i,\vec{z}_k)/\tau)}$, τ temperature controls how much to penalise hard negatives (lower temp penalises more) $\mathbf{1}_{[k \neq i]}$ evaluates to 1 iff $k \neq i$. Triplet Loss: where x is the image, x^+ is the positive image, and x^- is a random negative $\mathcal{L}_{triplet}(\vec{x},\vec{x}^+,\vec{x}^-) = \sum_{x \in \mathcal{X}}\max(0, ||f(x) - f(x^+)||_2^2 - ||f(x) - f(x^-)||_2^2 + \epsilon)$.
You evaluate the model with *Fine-tuning* (all params) or *linear probing* (freeze encoder). Batch size should be high (4096). Since large batch size is expensive: *BYOL*: (more robust to small batch-size) removes the need to use negative pairs in the loss function, instead optimize similarity on positive pairs. Needs new architecture (trivial solution where all inputs go to 0)



Prediction head in the student asserts "output of the student after the pred = output of target network without the pred.". It matches the output of online and target. DINO uses visual transformers as encoders and use the attention to visualise the attention map of the network when they create the self-supervised encoding; creates self-supervised segmentation maps.