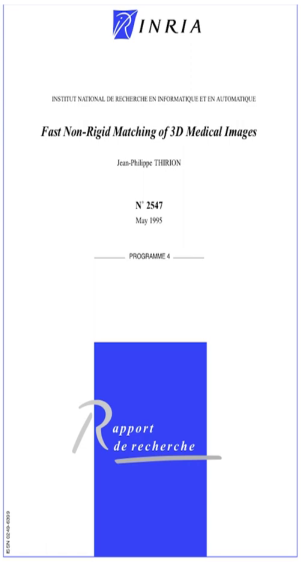
Medical Image Analysis
Professor Ganapathy Krishnamurthi
Department of Engineering Design
Indian Institute of Technology Madras

Lecture 19

Demons Part 1

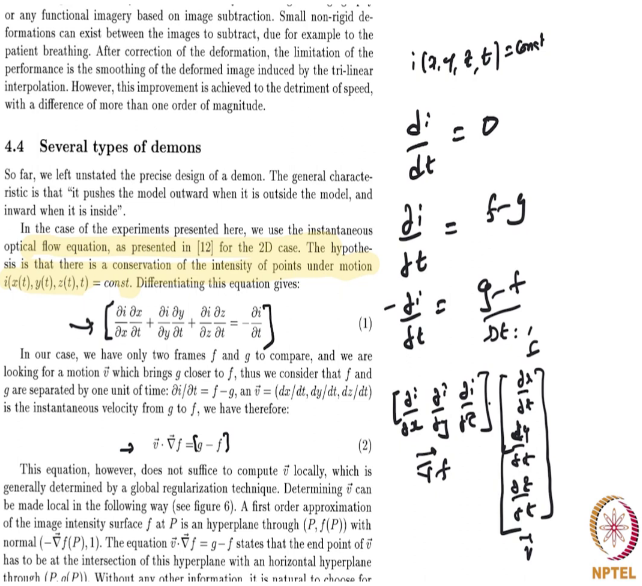
(Refer Slide Time: 00:15)



Hello, welcome back. So this video we are going to look at the nonrigid matching or registration algorithm called the demons registration by the lotto, or the author's name is Thirion is a very popular algorithm. It is been around for some time, and people still use it for nonrigid registration primarily because of ease of implementation.

And also, it is fairly quick algorithm, especially if you are trying to do 3d nonrigid registration of medical images. So, we will actually look at, how the deformations are computed first. And then we go back to why we this is called the demon's registrational algorithm.

(Refer Slide Time: 00:57)


or any functional imagery based on image subtraction. Small non-rigid deformations can exist between the images to subtract, due for example to the patient breathing. After correction of the deformation, the limitation of the performance is the smoothing of the deformed image induced by the tri-linear interpolation. However, this improvement is achieved to the detriment of speed, with a difference of more than one order of magnitude.

4.4 Several types of demons

So far, we left unstated the precise design of a demon. The general characteristic is that "it pushes the model outward when it is outside the model, and inward when it is inside".

In the case of the experiments presented here, we use the instantaneous optical flow equation, as presented in [12] for the 2D case. The hypothesis is that there is a conservation of the intensity of points under motion $i(x(t), y(t), z(t), t) = const$. Differentiating this equation gives:

$$\frac{\partial i}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial i}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial i}{\partial z}\frac{\partial z}{\partial t} = -\frac{\partial i}{\partial t} \quad (1)$$

In our case, we have only two frames f and g to compare, and we are looking for a motion $\vec{v}$ which brings g closer to f, thus we consider that f and g are separated by one unit of time: $\partial i/\partial t = f - g$, an $\vec{v} = (dx/dt, dy/dt, dz/dt)$ is the instantaneous velocity from g to f, we have therefore:

$$\vec{v} \cdot \vec{\nabla} f = g - f \quad (2)$$

This equation, however, does not suffice to compute $\vec{v}$ locally, which is generally determined by a global regularization technique. Determining $\vec{v}$ can be made local in the following way (see figure 6). A first order approximation of the image intensity surface f at P is an hyperplane through $(P, f(P))$ with normal $(-\vec{\nabla} f(P), 1)$. The equation $\vec{v} \cdot \vec{\nabla} f = g - f$ states that the end point of $\vec{v}$ has to be at the intersection of this hyperplane with an horizontal hyperplane through $(P, g(P))$. Without any other information, it is natural to choose for

So, in this paper under section 4.4, this is actually report, you are going to look at something called the optical flow based on which, the deformations or see individual pixel deformation fields are computed. So if you consider an image I.

And, and you say there are objects in the image at different locations x, y, z, and you also take this as a function of time, this is an optical flow, basically, you are trying to look at moving frames, that is, in a video, we are trying to talk track objects in a video etc, this optical flow is very useful.

So, the assumption is that the brightness of the object does not change between frames when I say brightness of the object, I mean, the pixel values comprising the object or the values of the pixels comprising the object does not change between frames. So, if that is true, then i is a constant, this is a constant for a particular image and we can say that $\frac{di}{dt} = 0$. So this is what is meant by, this condition here, one second right there.

consideration of intensity of points under motion, so, if I, if we are looking at video frame as an object is moving in the video frame, and then the pixels, which comprise the object, they do not change too much in intensity. And of course, the assumption is also that the motion between two successive frames are not large. So in our case, there is there are only two frames, basically the moving image and the fixed image, that is what we are going to look at until we will see how we can use this paradigm to arrive at a expression for the displacement field.

Now, this $\frac{di}{dt} = 0$ because i is a function of (x,y,z), certainly we can use chain rule to arrive at the following expression. So, which is it is given here, we use this, you understand expression. So, this $\frac{di}{dt} = 0$, we will translate to I using the chain rule in this form.

So, what is this $\frac{\partial i}{\partial t}$, $\frac{\partial i}{\partial t}$ is the if you discretize it, then we are just basically comparing the image intensity across two different frames, or in this case, image intensity difference between f and g, where f is the fixed image and g is the moving image. So, then we can write $-\frac{\partial i}{\partial t}$ = f - g. And then, of course, when there is a minus sign, we get to g - f.

The left hand side of this equation, it can be written as v.∇f, or we can see that very easily let me just, if you are not sure, let me just write it down so that you understand where this comes from. So, this left hand side can be written as $[\frac{\partial i}{\partial x} \frac{\partial i}{\partial y} \frac{\partial i}{\partial z}][\frac{\partial x}{\partial t} \frac{\partial y}{\partial t} \frac{\partial z}{\partial t}]$. So, this is basically something of it as a dot products. That is what is written on though this is your $\vec{v}$, and this is your $[\frac{\partial i}{\partial x} \frac{\partial i}{\partial y} \frac{\partial i}{\partial z}]$ is a 3 dimensional gradient vector. So, this gradient of f.

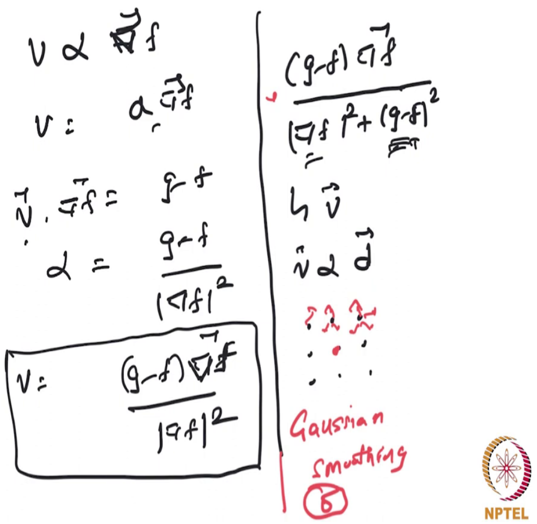
In this case, now I am using i, but then I know in the context that we are looking at f and g where f is, one frame and g is the other frame. This is just nothing but with respect to the fixed dimension ∇f. So, that is where you get this $v.\nabla f$ = g - f. It is actually $-\frac{\partial i}{\partial t} = \frac{g-f}{\partial t}$, and we set that equal to 1. So that is not explicitly shown here, because in order for the units to match on both since your v is the velocity, so the one way to understand this equation is that if you have, let us say we consider like a one dimensional a row of pixels, that is your image.

And if you have a slight motion of that row of pixels, and if you are taking the if you are looking at the difference between after the motion, if you are going to look at the difference between the pixel intensities, before and after motion, then you can say that this difference in intensity is equal to the speed at which the frame moves, multiplied by the rate at which the pixel values are changing.

So for a very small enough deformation, that is true, so then we can write it in this form. That is the explanation for this equation. So this is your optical flow equation, or this has a problem because it is determined. Because if you see v has three components. And in many cases, let us say if you are looking at 2d images, we have two components, but only one equation. Because we are actually trying to estimate v, because, in this case, we are looking at fixed frame and moving frame. And we are assuming that, $\partial t = 1$, so v is basically $\frac{dx}{dt}$, that is what we see here.

And we can just, we can always replace v with the displacement vector rather than the rather than use v. So, this v is what we are trying to estimate and so, it is not the equations is not number of equations is not enough. So, what this in this paper, what they did was or the author did was, he chose a regularization if you can call it or a constraint I should say on v, he just said that the author said that v is let me just add this.

(Refer Slide Time: 06:56)



So, $v \propto \nabla f$. So, the way I understand it, at least is that or some $v = \alpha . \nabla f$. If you assume this, then you can plug this back in the equation, we saw were in $v . \nabla f$ = g - f, if you substitute this, then you get the value of α to be substitute for instead of v substitute ($\alpha . \nabla f$). So, $\alpha = \frac{g-f}{|\nabla f|^2}$.

And then you put this, together α, you can always get $v = \frac{(g-f).\nabla f}{|\nabla f|^2}$ so, this is what we got. There are some problems with this equation, because it is it can be it can blow up because delta gradient of f is very small, you can get very large errors.

So, there is some kind of some form of a normalization or in this case, some you can call regularization of this expression, wherein instead of just using this, you have an additional term, so, you rewrite it as we rewrite it as $\frac{(g-f).\nabla f}{|\nabla f|^2+(g-f)^2}$. So, ∇f, sometimes it is small, it is offset by this number, which, in this case, it is very small, you still have this to, regularize the denominator so that it does not become too small and blow up. So this is the update. So this is v.

So if we have, so you can for each component of v, you can get this expression, we have a displacement field, of course, in the context of registration, v is basically proportional to your displacement small d, because we are looking at let us say a unit time interval, and we assume and then, so, instead of calling this velocity, we call this a displacement field. So, this is estimated.

On top of this, there is also a constraint to be, but not it is constrained, regularization to be done. So, why do we need this regularization? So, if you think of a grid? And if you are looking at the displacement vector, displacement vectors. So, think of it. There is no I mean, displacement vector can be in any direction. You are not constraining it at all. So this becomes this. So this is a very hard problem to solve either infinity of there is so many ways that you can displace your image and try to match the match it with the moving image the fixed image. So, there has to be some form of regularization.

So, in the original optical flow equation that you saw earlier, people used in Han Shan flow, people use some form of variational techniques to impose that regularization, by imposing smoothness constraints on the delivery on the velocity field. So, but in this case, they did a very simple regularization by doing a Gaussian, Gaussian smoothing.

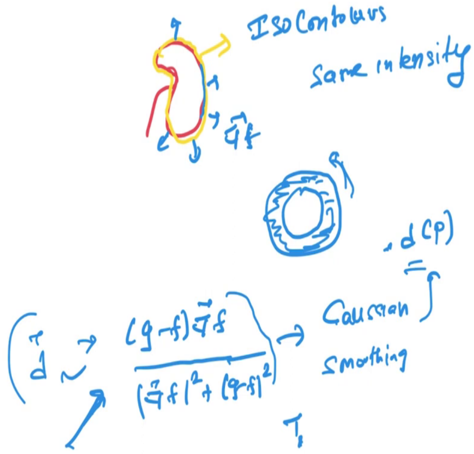
So, what does Gaussian smoothing do? All you do is you replace, if you take if based on this formula, you get an estimate for the for all the pixels in this grid. So, you replace for very crudely speaking, you replace the center value with the weighted average of the neighborhood. So, since this is a vector, so you can have average by average by component. So, you place every component with the average of the corresponding component in the neighborhood using a Gaussian weighting rather than uniform weighting use a Gaussian weighting, we saw that that is a Gaussian filter.

So, that is how it works. So the width of the filter sigma is a hyper parameter that would control to how much, you can call this a rigidity constraint. So, now you are kind of smoothing it so that, It is kind of uniform in a certain, it does not change very rapidly in a certain neighborhood, you get the same kind of displacement, and or smaller distances.

So, also note that, we are using the, we are trying to approximate v with using the direction of ∇f and the magnitude of the ∇f, you might ask, why do you why would we want to do that. So, one way of understanding is, is that, we want to move in the direction where there is more heterogeneity, where there is greatest change in the image.

So in the fixed said, image, ∇f tells you the direction of greatest changes, heterogeneous regions, that is where we want them, we want to move in those directions, because wherever good enough is very small, it means it is a very flat region, and you really do not have any information there. So, that is, that is one thing that helps if you move in a direction of ∇f. Another thing is, if you look at images, so for instance, if you look at boundaries of organ. So let us go to another slide.

(Refer Slide Time: 12:25)



So let us go to let us say we have a kidney, let us say something like this, if you look at its boundary or body of any organ for that matter. So this you can interpret as a ISO contour, ISO contour, or regions, or lines have similar intensity or same intensity. So, if you take a gradient, the gradient is always perpendicular to the direction of the ISO contours.

If you take any image or ∇f, if this is the image, then the gradient is always perpendicular to ISO contour. And this is good, because, it is very hard, but near impossible to, to figure out motions along regions, along the lines where intensity does not change.

So let us say we are perfectly just to give an example of why that is the case, where you cannot determine anything along regions where there is no change in intensity, or $\nabla f = 0$, basically, that is too great of f would be 0 along the along the contour along the edges. So, if you have any kind of displacement along the edges anyway, it is going to be impossible to figure out.

So, you can think of an example like if you have a disk. And it is of uniform color here, let us say this is all blue if it is and if it is spinning, let us say it is it is rotating you do not know. If it is a very solid same same color, only then there are variations in the intensity on the face of this disc, only then can you see it spinning, otherwise, it is impossible for the spin.

So similarly, it is the same case if you have, no variations along the case of contours basically along the edges you might not know or where or the $\nabla f = 0$, you cannot estimate any displacement. So, then you only look in directions which are perpendicular twice or contours or along the direction of the gradient. That makes more sense. So that is why you put your $v \propto \nabla f$ is one explanation you can think of.

The author in this paper gives some geometric explanations, but I have given you this you can also read the paper to understand the geometry explanation. So, this is the overall idea. So, there are there is a first step is you estimate the without the displacement field, and then you regularize it by smoothing with a Gaussian filter.

The regularization is an essential step otherwise, because, the problem is impossible, you have so many directions in which you can move and, unless you constrain it, you cannot solve it. It will blow up very quickly your solutions. So, and in the in the paper, the authors suggest multiple variants of the formula that we saw, to estimate the displacements. So, it is all given there. So please go look at that, but this is the one I showed you was the standard one.

So, that is usually done, so that is to rewrite it, you would estimate this as $\frac{(g-f).\nabla f}{|\nabla f|^2+(g-f)^2}$, so this is proportional, something that you would do for the displacement field. So this, is a very simple and efficient technique, this followed by Gaussian smoothing. That is the two steps for the algorithm.

So here, this is an iterative algorithm. So what you would do is you would first start with an identity transformation, when you do not do anything, you start with the fixed and the moving image, use this formula to estimate let us say, T1 transformation 1, this transformation is nothing but a bunch of displacement fields for your image here, you would consider all the pixels in the image so which means that you would estimate at d for every pixel. So, d for every pixel in the image, p refers to the position of every pixel.

So remember, we always do registration in physical coordinates, so p represents the physical coordinates of the pixels in the moving image. So you estimate for every pixel in the moving image, a corresponding v of p. And while you can do either way, so there is a inverse transform also, so it is up to you.

So, you have a displacement field that you figured out for every pixel. And you smooth it, you smooth it with the Gaussian and after smoothing, and you apply it to the moving image, and then come back to this step. So, it is like, and once you do that, every time and every time you want to update the transformations, so, initially the identity transformation, now, you have to update that identity transformation with this.

So, then you have a new transformation after this, once again, you deform the image and then perform this step, whatever you are seeing here, followed by Gaussian smoothing, and then you go ahead and update the deformation field or the transformation. So, you keep doing that till a fixed number of iterations and then install.

So, this is the typical algorithm for the demons registration, very simple technique, computationally and so very popular and lot of analysis has been done now, this is equivalent to minimizing a certain type of loss function, if you want to do gradient descent on a loss function or estimate the d. Or there are papers which talk about how you can interpret it that way, but from the point of view of optical flow techniques this is the easiest interpretation. So, the author calls it demons. Now, we will see why he calls the demons, shortly.