Spatial Transformer Spectral Kernels for Deformable Image Registration

Ebrahim Al Safadi
alsafadi@ohsu.edu
Xubo Song
songx@ohsu.edu

Center for Spoken Language Understanding
Oregon Health & Science University
Oregon, USA

Abstract

Recent advances in kernel methods have made them more attractive tools for spatial transformation models. In this work, Spatial Transformer Spectral Kernels are introduced as a framework for Deformable Image Registration. The transformation is restricted to live in a Reproducing Kernel Hilbert Space, and Generalized Spectral Mixture kernels are used as the reproducing kernels. This combination results in a powerful but simple regularization model that can adapt to many deformation scenarios with nonstationary and possibly long range nonmonotonic relations across the pixels. Our formulation leads to a Kernel Ridge Regression transform that is pre-computed once before optimization, and unlike most developments in image registration the loss function explicitly pairs this transform with a specific interpolation function. We derive a closed-form gradient of the loss function with respect to the spatial transformation and interpolation function which enhances registration results. Based on our evaluation, while being simpler our method can perform comparably to more complex Large Deformation Diffeomorphic Metric Mapping models in terms of reducing the intensity sum of squared differences, and can provide a more accurate estimate of the underlying displacement field.

1 Introduction

Deformable image registration is the process of finding a nonlinear spatial transform of an image that brings it into spatial alignment with another image [23]. Although it has many applications in computer vision, image processing, and medical imaging, it remains an ill-posed problem that is hard to generalize.

On the other hand, the theory of reproducing kernels provides powerful models for regularized and highly adaptive transformations. It enables describing a transformation in a Reproducing Kernel Hilbert Spaces (RKHS) as a finite sum of coefficients scaled by the reproducing kernels [14, 19]. Furthermore, it simplifies the difficult regularization process that involves penalizing norms of differential operators. As such, it has received a lot of attention recently as a framework for spatial transformations. For example, RKHS has been used to parameterize velocity fields in Large Deformation Diffeomorphic Metric Mappings (LDDMM) within the field of Computational Anatomy [6, 12], and to model discontinuous transformations for medical imaging tasks with sliding organs [9].

Kernel methods generally scale poorly with data size, and therefore were considered in image registration mostly in the context of landmark matching [23, 24], and to a limited

extent in intensity-based methods [16]. However, the wealth of recent advances in kernel methods necessitates reconsidering them in such applications to exploit their powerful regularization properties. The RKHS framework is extremely rich, and allows for sparse and finite support kernels [9], nonstationarity and spectral confinement [15], and a variety of low-rank approximations [10, 25].

In this work, we propose a deformable image registration method that models the spatial transformation as an additive displacement function that lives in RKHS. To maximize the adaptability of the transformation while retaining the special properties of the RKHS framework, the reproducing kernel is chosen as a Generalized Spectral Mixture (GSM) kernel [15]. This is a recently-introduced kernel that permits spatially varying kernel parameters while regulating their spectral content. Furthermore, our formulation of the optimization problem for the displacement model leads to a Kernel Ridge Regression (KRR) transform [14] that is pre-computed once before optimizing the displacement field.

To limit computational complexity, the dense displacement field is parameterized by displacements off of a sample of spatial points. Furthermore, particular emphasis is made on the re-sampling (interpolation) stage in this work. Whereas image registration papers typically treat the re-sampling step as an afterthought, and don't explicitly mention it in the loss function, it is given an explicit and generalized form in this work. As such, a closed form for the gradient of image registration loss function is evaluated which takes into account the KRR transformation and interpolation function.

The formulation in this work places it in a position between simpler, more rigid methods such as Free-Form-Deformations (FFD) [18, 20], and more complex LDDMM methods [3, 22]. Whereas FFDs are generally confined to a fixed grid of control points, and use specific local functions such as B-Splines, our method can sample from an arbitrary set of points, vary its parameters across space, and choose from a wealth of composite kernels enabled by the properties of RKHS. LDDMM methods, on the other hand, focus on preserving topology and optimizing the flow of a time-dependent vector field. Numerical integration is usually needed to find the path of diffeomorphisms which makes these methods computationally intensive [12].

The closest work to the one proposed here can be found in [8, 9]. The authors there defined an RKHS to model a displacement field, and proved a new Representer theorem that included an ℓ_1-norm penalty on the transformation to promote sparsity. However, their emphasis was on addressing transformations with discontinuities in particular to tackle the problem of sliding organs, and used a different formulation than the KRR transform derived here. They also used Wendland kernels to limit the support of the transformation and to operate better with discontinuous transformations.

2 Method

2.1 Problem formulation

Registering a moving image J to a fixed image I requires two operations on J, namely, spatial transformation of its pixel locations followed by interpolation of the transformed image at a common pixel grid. Assume image $\mathrm{J} : \mathcal{X} \to R$ is a mapping between a d-dimensional pixel grid $\mathcal{X} \in R^d$ and intensity values in the reals. A spatial transformation $\mathcal{T}(\mathcal{X})$ operates on the pixel grid but does not in itself change the intensity values of the transformed image.

To measure the alignment between the transformed image $\mathrm{J}(\mathcal{T}(\mathcal{X}))$ and the fixed image

I, the two are compared at a common pixel grid, and therefore require interpolating the transformed image back at $\mathcal{X}$. Given a general interpolation function Ψ, if an arbitrary function has values $y = (y_1, \cdots, y_n)$ at spatial points $x = (x_1, \cdots, x_n)$, then $y' = \Psi(y, x, x')$ denotes the output of the interpolator at new points x'. This function will be restricted to dot products of the form

$$y'_i = \sum_{k \in \mathcal{N}_i} y_k \psi(x_k, x'_i), \tag{1}$$

where ψ is an interpolation kernel and $k \in \mathcal{N}_i$ indexes a small neighborhood of points around x_i. This is sufficient to cover a wide variety of interpolation methods such as nearest neighbors, linear, cubic, and kernel smoothing. The transformed and resampled image can then be written as

$$\tilde{\mathrm{J}} = \Psi(\mathrm{J}, \mathcal{T}(\mathcal{X}), \mathcal{X}), \tag{2}$$

which will be used to denote the new intensity values of J if its pixels were deformed to $\mathcal{T}(\mathcal{X})$ and then it was interpolated at $\mathcal{X}$. The interpolation function Ψ is assumed to be fixed albeit with a parameterizable interpolation kernel $\psi(x, x'; \vartheta)$.

The transformation model $\mathcal{T}(\mathcal{X})$ is assumed to be additive, i.e., $\mathcal{T}(\mathcal{X}) = \mathcal{X} + \mathcal{U}(\mathcal{X})$, where $\mathcal{U}(\mathcal{X})$ is a displacement field. The objective is then to solve the optimization problem

$$\begin{aligned} \mathcal{U}^* &= \underset{\mathcal{U} \in \mathcal{R}^n}{\arg\min}\, \mathcal{L}\big(\mathrm{I}, \tilde{\mathrm{J}}(\mathcal{U})\big) \\ &= \underset{\mathcal{U} \in \mathcal{R}^n}{\arg\min}\, \mathcal{L}\big(\mathrm{I}, \Psi(\mathrm{J}, \mathcal{X} + \mathcal{U}(\mathcal{X}), \mathcal{X})\big), \end{aligned} \tag{3}$$

for some loss function $\mathcal{L}$.

Assume the displacement field $\mathcal{U}$ is known to have values $\{\theta_i\}_{i=1}^m$ at an arbitrary set of points $\mathcal{G} = (g_1, g_2, \ldots, g_m)$, which could be, for instance, a grid of control points. A function $\mathcal{F}$ is sought that will generalize from these known samples to define $\mathcal{U}$ at any location. Furthermore, the function is restricted to live in a RKHS, $\mathcal{H}$, such that

$$\mathcal{F}^* = \underset{\mathcal{F} \in \mathcal{H}}{\arg\min} \sum_{i=1}^{m} \big(\theta_i - \mathcal{F}(g_i)\big)^2 + \sigma \|\mathcal{F}\|^2_{\mathcal{H}}. \tag{4}$$

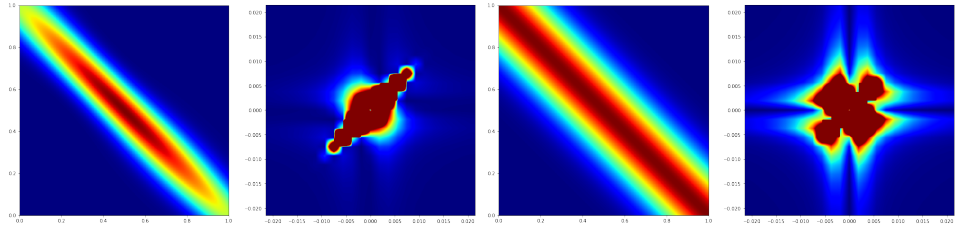
Applying Representer Theorem [14, 19], the optimum transform will admit a finite and linear form $\mathcal{F}^*(\cdot) = \sum_{i=1}^m \alpha_i k(\cdot, g_i)$ for some real constants α (to be found), where $k(\cdot, g_i)$ is a symmetric positive definite kernel that reproduces the function space $\mathcal{H}$ by the Moore-Aronszajn theorem. For compactness, $k(\cdot, \mathcal{G}) \doteq \big(k(\cdot, g_1), \ldots, k(\cdot, g_m)\big)$, and the displacement at any pixel point $p_j \in \mathcal{P} \subset \mathcal{X}$ can be written as $u_j = \alpha^T k(p_j, \mathcal{G})$, where $\mathcal{P}$ is a pixel grid.

Substituting $\mathcal{F}^*(\cdot) = \alpha^T k(\cdot, \mathcal{G})$ into (4) and solving for α returns a Kernel Ridge Regression form, $\alpha = \big(K_{\mathcal{G},\mathcal{G}} + \sigma \mathrm{Id}\big)^{-1} \theta$. Consequently, the displacement of all the pixels can be compactly written as

$$\tilde{\mathcal{P}} \doteq \mathcal{T}(\mathcal{P}) = \mathcal{P} + K_{\mathcal{P},\mathcal{G}} \big(K_{\mathcal{G},\mathcal{G}} + \sigma \mathrm{Id}\big)^{-1} \theta, \tag{5}$$

where $\big[K_{\mathcal{P},\mathcal{G}}\big]_{i,j} = k(p_i, g_j)$. Finally, plugging the transformation model (5) into the registration problem (3), with the spatial points restricted to the pixels $\mathcal{P}$, regularizes the displacement field and reduces the problem in (3) to the simpler problem

$$\theta^* = \underset{\theta \in \mathcal{R}^m}{\arg\min}\, \mathcal{L}\big(\mathrm{I}, \Psi(\mathrm{J}, \mathcal{P} + K_{\mathcal{P},\mathcal{G}} \big(K_{\mathcal{G},\mathcal{G}} + \sigma \mathrm{Id}\big)^{-1} \theta, \mathcal{P})\big). \tag{6}$$



(a) (b) (c) (d)

Figure 1: Comparison between GSM and RBF kernels (a) Spatial profile of one realization of a non-stationary GSM Kernel with a single spectral mixture component. (b) PSD profile of this GSM Kernel. (c) RBF Kernel spatial profile. (d) PSD profile of RBF Kernel.

There are many advantages to this formulation. First of all, it reduces the transformation to a linear operation $M \doteq K_{\mathcal{P},\mathcal{G}}\left(K_{\mathcal{G},\mathcal{G}} + \sigma\,\mathrm{Id}\right)^{-1}$ on a small set of points, θ, and this matrix is pre-computed once *before* plugging it into (6) and finding an optimum displacement field. It can also be approximated by a variety of recent low-rank approximation techniques [10, 25]. Secondly, this form involves the interpolation function Ψ in the optimization process and helps reduce interpolation artifacts that commonly arise in image registration problems [2, 13, 17, 21]. Lastly, this formulation enables leveraging a wealth of modern kernel methods. To make the transformation model non-stationary and spectrally-regularized, a GSM kernel is used which takes the form

$$k_{GSM}(x,x') = \sum_{i=1}^{Q} w_i(x) w_i(x')\, k_{Gibbs,i}(x,x') \cos(2\pi(\mu_i(x)x - \mu_i(x')x')), \tag{7}$$

where $w_i(x)w_i(x')$ captures the similarity between x and x' based on their combined variance,

$$k_{Gibbs}(x,x') = \sqrt{\frac{2\ell_i(x)\ell_i(x')}{\ell_i^2(x)+\ell_i^2(x')}} \exp\left(\frac{-(x-x')^2}{\ell_i^2(x)+\ell_i^2(x')}\right), \tag{8}$$

denotes a nonstationary Gibbs kernel with spatial function $\ell(x)$, and the function $\mu(x)$ controls the spectral surface of the kernel [15]. Despite their composite form, like Radial Basis Functions (RBFs) they can also be decomposed for a point in $D-$dimensions as $k_{GSM}(g_i, g_j|\vartheta) = \prod_{d=1}^{D} k_{GSM}(g_i^d, g_j^d|\vartheta_d)$, for a parameter set ϑ so that $K = K_x \otimes K_y \otimes \cdots \otimes K_D$. Fig. 1 shows a comparison between the spatial and Power Spectral Density (PSD) profiles of RBFs and a simple GSM with a single spectral mixture component. Notice how the GSM kernel in (a) returns different values for different pairs (x,x') even if the distance between them is similar, unlike in (c), and how we concentrated the frequency support to have a tighter bandwidth in (b) compared to (d). Whereas here we have only biased the kernel to increase elasticity near the center of the image, in its general setting the GSM kernel can represent non-local and non-monotonic dependencies and therefore model more complicated relations between x and x'.

2.2 Optimization

Given the optimization problem in (6), the gradient of the loss function with respect to θ, $\nabla_\theta \mathcal{L}$, will be evaluated in closed form. To this end, the chain rule involves partial derivatives of the loss function $\mathcal{L}$, the interpolation function Ψ, and lastly the displaced pixels $\tilde{\mathcal{P}}$ with respect to θ.

The last step is simplified by the linearity of the kernel transformation model $M\theta$ so that $\partial\tilde{\mathcal{P}}_i/\partial\theta_j$ can directly pass through M. Consequently, the derivative of the loss with respect to θ_j can be computed as

$$\big[\nabla_\theta \mathcal{L}(\theta)\big]_j = \frac{\partial \mathcal{L}}{\partial \theta_j} = \sum_{i=1}^{n} \frac{\partial \mathcal{L}}{\partial \tilde{\mathrm{J}}_i} \frac{\partial \tilde{\mathrm{J}}_i}{\partial \theta_j}. \tag{9}$$

Assuming pixels have been transformed by a KRR transformation $M\theta$ followed by a generic interpolator ψ, (1) becomes

$$\tilde{\mathrm{J}}_i = \sum_{k\in\mathcal{N}_i} \mathrm{J}_k \psi(p_i, \tilde{p}_k(\theta)). \tag{10}$$

As such, the derivative with respect to θ_j becomes

$$\begin{aligned}
\frac{\partial \tilde{\mathrm{J}}_i}{\partial \theta_j} &= \frac{\partial}{\partial \theta_j} \sum_{k\in\mathcal{N}_i} \mathrm{J}_k \psi(p_i, \tilde{p}_k(\theta)) \\
&= \sum_{k\in\mathcal{N}_i} \mathrm{J}_k \frac{\partial}{\partial \theta_j} \psi(p_i, p_k + \sum_n \theta_n m_{k,n}) \\
&= \sum_{k\in\mathcal{N}_i} \mathrm{J}_k \dot{\psi}\big(p_i, \tilde{p}_k(\theta)\big) \big(\sum_n m_{k,n} \frac{\partial}{\partial \theta_j} \theta_n\big) \\
&= \sum_{k\in\mathcal{N}_i} \mathrm{J}_k \dot{\psi}\big(p_i, \tilde{p}_k(\theta)\big) m_{k,j},
\end{aligned} \tag{11}$$

where $\dot{\psi}$ denotes the derivative of ψ. For example, if $\psi(p_i, p_j) = \exp\big(-\|p_i - p_j\|^2/\ell\big)$, then the derivative for the x^{th}-coordinate is

$$\dot{\psi}\big(p_i^x, \tilde{p}_k^x(\theta)\big) = \frac{-2}{\ell} \psi\big(p_i^x, \tilde{p}_k^x(\theta)\big)\big(p_i^x - \tilde{p}_k^x(\theta)\big) \tag{12}$$

and (11) becomes, again for the x^{th}-coordinate,

$$\frac{\partial \tilde{\mathrm{J}}_i}{\partial \theta_j^x} = \frac{-2}{\ell} \sum_{k\in\mathcal{N}_i} \mathrm{J}_k \psi\big(p_i^x, \tilde{p}_k^x(\theta)\big)\big(p_i^x - \tilde{p}_k^x(\theta)\big). \tag{13}$$

The complexity of this formulation lies in evaluating the gradient, which can run in two modes: a) The full mode which evaluates the effect of each of the m control points on the n pixels and requires $n \times m$ entries, and b) The fast mode which only evaluates derivatives $\partial\tilde{\mathrm{J}}_i/\partial\theta_j$ for nearby pixels and control points and could drop to n evaluations. This can be done without significant loss in performance with kernels of local support such as Wendland kernels [12].

2.3 Closed-form gradient for a Nadaraya-Watson (NW) interpolator

The interpolator form was observed to give significantly different results for the same spatial transformation to warrant additional details. A fairly generic and easily parameterizable interpolation function is the Nadaraya-Watson interpolator which, when adjusted for notation, gives the form [5]

$$\tilde{\mathrm{J}}_i = \frac{\sum_{k\in\mathcal{N}_i} \mathrm{J}_k\, k(p_i, \tilde{p}_k(\theta))}{\sum_{k\in\mathcal{N}_i} k(p_i, \tilde{p}_k(\theta))}, \tag{14}$$

where the spatially invariant interpolation kernel $k(\cdot,\cdot)$ is chosen as an RBF kernel with spread ℓ_{NW}.

The support of this kernel interpolator is strictly local and controlled by $k_{NW}-$nearest neighbors $\mathcal{N}_i$ around a point x_i, and in the basic RBF setting has two additional parameters which are the kernel spread ℓ_{NW} and k_{NW}. These could be tuned or optimized depending on the type of images to be aligned. To treat the normalization factor in (14), let

$$f_i(\theta_j) = \sum_{k\in\mathcal{N}_i} \mathrm{J}_k k(p_i, \tilde{p}_k(\theta)) \doteq \sum_{k\in\mathcal{N}_i} \mathrm{J}_k \alpha_k, \tag{15}$$

and

$$g_i(\theta_j) = \sum_{k\in\mathcal{N}_i} k(p_i, \tilde{p}_k(\theta)) \doteq \sum_{k\in\mathcal{N}_i} \alpha_k. \tag{16}$$

Taking their derivatives with respect θ_j gives, respectively,

$$f_i'(\theta_j) = \sum_{k\in\mathcal{N}_i} \mathrm{J}_k k(p_i, \tilde{p}_k(\theta)) \frac{-2}{\ell_{NW}} (p_i - \tilde{p}_k(\theta)) m_{k,j} \doteq \sum_{k\in\mathcal{N}_i} \mathrm{J}_k \beta_k, \tag{17}$$

and

$$g_i'(\theta_j) = \sum_{k\in\mathcal{N}_i} k(p_i, \tilde{p}_k(\theta)) \frac{-2}{\ell_{NW}} (p_i - \tilde{p}_k(\theta)) m_{k,j} \doteq \sum_{k\in\mathcal{N}_i} \beta_k. \tag{18}$$

Combining these and using the quotient rule results in the partial derivatives

$$\frac{\partial \tilde{\mathrm{J}}_i}{\partial \theta_j} = \frac{(\sum_k \mathrm{J}_k \beta_k)(\sum_k \alpha_k) - (\sum_k \mathrm{J}_k \alpha_k)(\sum_k \beta_k)}{(\sum_k \alpha_k)^2}, \tag{19}$$

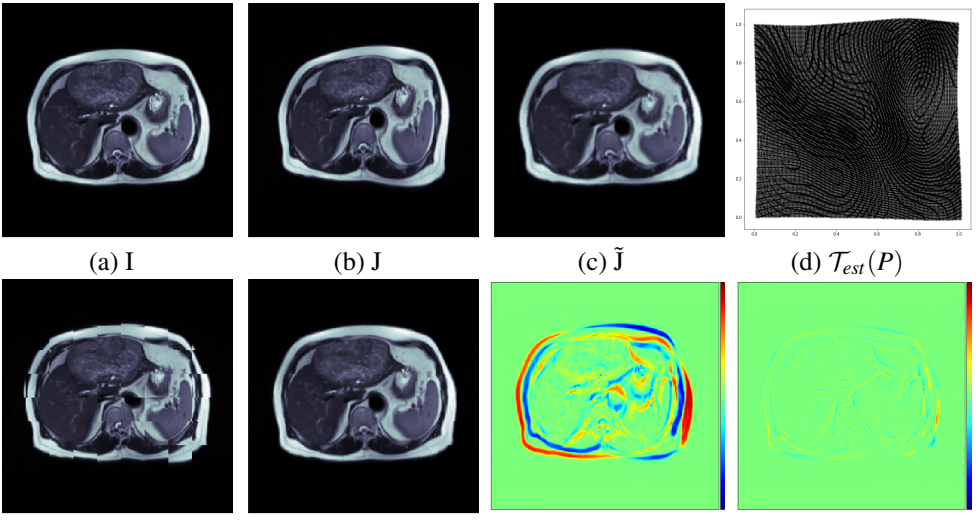
where $\alpha_k \doteq k(p_i, \tilde{p}_k(\theta))$, and $\beta_k \doteq \frac{-2}{\ell_{NW}} k(p_i, \tilde{p}_k(\theta))(p_i - \tilde{p}_k(\theta)) m_{k,j}$. Substituting (19) into (9) gives the final form of a gradient element when a generic spatial transform in RKHS is combined with an NW-interpolator. This could be used in any gradient-based or quasi-Newton method for optimization.

3 Experiments and Results

Two experiments were conducted to test the method outlined in this paper. For the first one, we applied our method to an MRI image of Fibrolamellar hepatocellular carcinoma (liver), which was artificially deformed with a spatio-temporal field to generate a sequence of images with known ground truth. For the second experiment, we used a pair of MRI brain images of a patient with progressing Alzheimer's Disease taken two years apart.

Since the outlined method has a variety of realizations depending on the spatial transform reproducing kernel and interpolation kernel, two of each kernels are used and the four combinations are tested. The two reproducing kernels for spatial transformations are RBF, which is spatially invariant, and GSM, which is nonstationary and spectrally regularized, and which was biased to have more elasticity towards the center of the image. We stress that this choice is mostly illustrative and that alternative profiles are diverse and learnable [15]. The two interpolation kernels are Nearest Neighbors (NN) which is nondifferentiable, and Nadaraya-Watson (NW) which is smooth and parameterizable.

Furthermore, these variations were compared with four popular registration methods: Rigid, Affine, FFD with B-Spline, and a more powerful diffeomorphic method, namely, Symmetric Normalization (SyN) [3].



(a) I (b) J (c) $\tilde{\mathrm{J}}$ (d) $\mathcal{T}_{est}(P)$

(e) Checker(I,J) (f) Checker(I,$\tilde{\mathrm{J}}$) (g) I − J (h) I − $\tilde{\mathrm{J}}$

Figure 2: Image registration results for the fourth pair of liver images. (a) Fixed image. (b) Moving image. (c) Moving image after registration. (d) Estimated spatial transformation. (e) Checkerboard of image pair before registration. (f) Checkerboard of image pair after registration. (f) Heat map of the difference between images before registration. (g) Heat map after registration.

FFD was implemented with SimpleITK [26], and SyN was implemented in ANTsPy [1]. Our method was implemented in Python with the interpolation and gradients written in Cython. For all the methods we fixed the loss function to Sum of Squared Differences (SSD) and used L-BFGS as an optimizer. Although the method in this paper can be used in a hierarchical fashion, progressing from coarse to finer grids, this was avoided here to measure baseline performance. The grid density, which forms θ, was fixed to 10×10 throughout these results. Moderate fine tuning of kernel parameters was done to select a reasonable parameters for σ and the NW-interpolator which was done once on the synthetic data and fixed afterwards.

3.1 Evaluation using liver image with known ground truth

To accurately assess the ability of the different algorithms to both reduce the intensity loss and correctly capture the underlying displacement field, a synthetic spatio-temporal field was generated to resemble one breathing cycle. Eight snapshots of spatial fields were sampled at equal time intervals from

$$\mathcal{U}(x,y;t) = \alpha \sum_{i=1}^{5} \mathcal{N}(x,y;\mu_i(t),\Sigma_i(t)) + \beta \sin(2\pi(t+\phi)) \sin(\pi(x+y)), \tag{20}$$

where α, β, and ϕ were set to 0.02, 0.025, and 0.2, respectively (with x, y, and t normalized to $[0,1]$). The five covariance matrices were given different angles and eccentricities. This field was applied to an MRI image of a liver with a large heterogeneous mass in the left lobe [2], generating a sequence of 8 images to be registered for motion correction. See Fig. 2.

Method	$\mathcal{U}$-SSD	$\mathcal{U}$-NCC	Intensity RSSD	Intensity INCC
Rigid	1.0784 ± 0.0806	34.98% ± 9.62%	0.7877 ± 0.1538	6.47% ± 4.24%
Affine	1.0612 ± 0.2863	58.26% ± 12.81%	0.3948 ± 0.0663	19.97% ± 8.15%
FFD-BSpline	0.4137 ± 0.0576	76.96% ± 3.42%	0.0214 ± 0.0066	31.66% ± 10.87%
KRR-RBF-NN	0.3849 ± 0.2007	80.20% ± 10.72%	0.1032 ± 0.0314	29.36% ± 11.18%
KRR-RBF-NW	0.3542 ± 0.1630	80.12% ± 10.26%	0.0318 ± 0.0081	31.45% ± 11.18%
KRR-GSM-NN	**0.3427 ± 0.1690**	81.72% ± 9.48 %	0.1080 ± 0.0755	29.29% ± 11.36%
KRR-GSM-NW	0.3580 ± 0.2312	**82.14% ± 11.05%**	0.0435 ± 0.0134	31.11% ± 11.23%
SyN	0.4980 ± 0.1939	79.59% ± 8.42%	0.0155 ± 0.0052	31.94% ± 11.25%

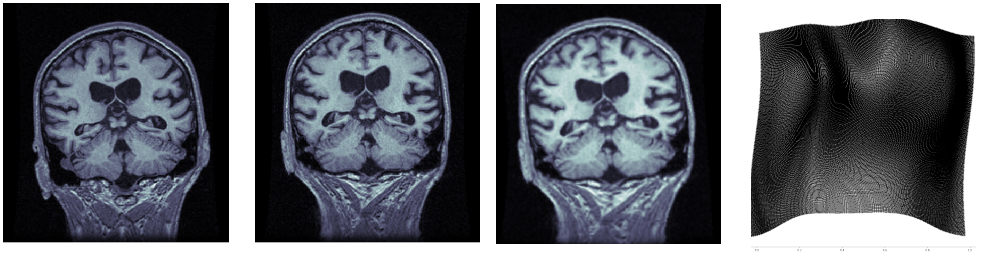
Table 1: Comparison of registration Methods using four metrics on a sequence of liver images with known displacement.

Method	Final SSD	RSSD	NCC	INCC	$\sigma_{\text{dct}}(D_x)$	$\sigma_{\text{dct}}(D_y)$
Rigid	803.5321	0.4037	88.10%	25.54%	-	-
Affine	726.5577	0.3650	89.36%	27.35%	0.0206	0.0367
FFD-BSpline	761.6180	0.3826	88.41%	25.99%	0.0135	0.0173
KRR-RBF-NN	764.6149	0.3841	88.93%	26.73%	0.0172	0.0288
KRR-RBF-NW	657.5258	0.3303	90.16%	28.48%	0.0181	0.0315
KRR-GSM-NN	820.8827	0.4124	88.09%	25.53%	0.0114	0.0255
KRR-GSM-NW	635.9846	0.3195	90.49%	28.96%	0.0148	0.0268
SyN	606.0322	0.3044	91.30%	30.07%	0.0219	0.0402

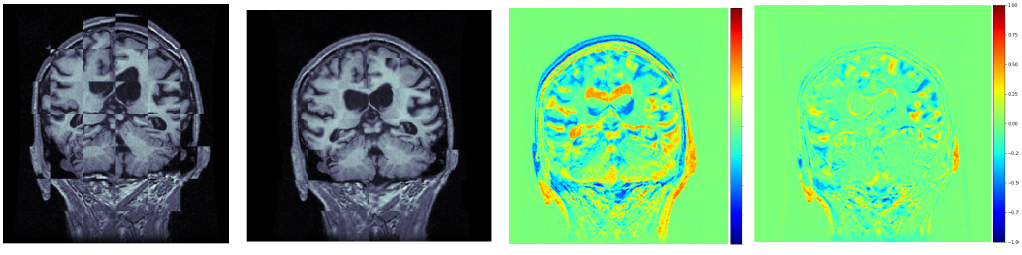
Table 2: Results for brain images. Original SSD: 1990.6303, Original NCC: 70.17%.

To compare the results, several metrics were used. The displacement SSD, defined as $\mathcal{U}\text{-SSD} = \|\mathcal{U}^{est} - \mathcal{U}^{true}\|^2 / \|\mathcal{U}^{true}\|^2$ for known and estimated displacement fields $\mathcal{U}^{true}$ and $\mathcal{U}^{est}$, respectively, was computed for each pair. Similarly, the displacement NCC was computed as $\mathcal{U}\text{-NCC} = \rho(\mathcal{U}^{true}, \mathcal{U}^{est})$. We also calculated $\text{RSSD} = \|\text{I} - \tilde{\text{J}}\|^2 / \|\text{I} - \text{J}\|^2$ (the reduction in intensity SSD) and $\text{INCC} = \rho(\text{I}, \tilde{\text{J}}) / \rho(\text{I}, \text{J}) - 1$ (the increase in intensity NCC) as a result of registration. We listed the means and standard deviations of these values in Table 1.

From an intensity matching side, the proposed methods outperform non-deformable transformations by wide margins and perform comparably to FFD and SyN in terms of average increase in NCC (all between 31-32%). The more complex LDDMM method, SyN, outperformed our methods in terms of average reduction in SSD as expected. However, when comparing to the ground truth spatial transformations our methods consistently outperformed, hinting to the possibility that the LDDMM methods, even after tuning their regularization parameters, could still risk overfitting. Interestingly, this was also independently observed in [22], where it was noted that LDDMM methods could create large deviations between landmarks, even when they successfully reduce the intensity SSD. Furthermore, within our methods, the simple GSM kernel used provided a boost in Grid-NCC by roughly 2% over RBF. There is room for improvement once a more elaborate use of this kernel is developed.



(a) I (b) J (c) $\tilde{\mathrm{J}}$ (d) $\mathcal{T}_{est}(P)$



(e) Checker(I, J) (f) Checker(I, $\tilde{\mathrm{J}}$) (g) I − J (h) I − $\tilde{\mathrm{J}}$

Figure 3: Registering a pair of brain images with progressing atrophy. (a) Fixed image. (b) Moving image. (c) Moving image after registration. (d) Estimated spatial transformation. (e) Checkerboard of image pair before registration. (f) Checkerboard of image pair after registration. (f) Heat map of the difference between images before registration. (g) Heat map after registration.

3.2 Evaluation using brain image

The second experiment is done on a recent data set called Minimal Interval Resonance Imaging in Alzheimer's Disease (MIRIAD) [□]. It includes MRI scans of patients with Alzhiemer's Disease taken over 10 sessions over a period of two years, and provides a valuable source for tracking brain atrophy over relatively close intervals. The data of subject 204 was used with the fixed and moving images taken from the first and last MRI sessions of that subject, respectively (See Fig. 3). The difference between I and J includes both rigid and deformable components, with the latter due to anatomical changes caused by brain atrophy.

The intensity RSSD and INCC were computed. In addition, to measure the spectral characteristics of the transformation, 2D Discrete Cosine Transforms (DCT) of the x and y-components of the displacement fields were computed and the standard deviations $\sigma_{\mathrm{dct}}(D_x)$ and $\sigma_{\mathrm{dct}}(D_y)$ were reported. These statistics measure the overall smoothness of the displacement field along the x and y axes, and were generally observed to correlate negatively with RSSD. See the results in Table 2.

The RSSD of the KRR model with GSM as a kernel and NW as an interpolator performed the best among our methods, reducing the final SSD to 31.95% of its original value and outperforming FFD. It did so while maintaining a spectral spread bellow SyN and the variations we used with RBF. However, SyN reduced the SSD to 30.44% and was easier to tune for a trade-off between RSSD and displacement smoothness as indicated by σ_{dct}.

4 Conclusion

In this work a method for deformable image registration was introduced with a spatial transformation that lives in a RKHS and takes the form of Kernel Ridge Regression. The reproducing kernel was chosen as a Generalized Spectral Mixture (GSM) kernel to regulate the spectral content of the transformation while being spatially variant. Controlled experiments show the methods not only perform reasonably well at reducing the dissimilarity of the image intensities, but can also provide more accurate estimates of the underlying displacement field on our test data than more complex transformation models such as LDDMM which demonstrated a higher risk of overfitting. Closed form formulas for the gradient through the spatial transformer and generic interpolator were also derived, providing interpolation-aware optimization of the spatial transformation.

The results obtained are promising and encourage expanding this work to deep architectures and LDDMM settings. For example, Spatial Transformer Networks [7] have become popular layers in deep models and use affine transformations. The KRR model derived here could be an interesting advancement on that model.

References

[1] Advanced normalization tools in python. https://github.com/ANTsX/antspy.git, 2019.

[2] I. Aganj, B. Yeo, M. R. Sabuncu, and B. Fischl. On removing interpolation and resampling artifacts in rigid image registration. *IEEE Trans. on Image Processing*, 22(2): 816–827, Feb. 2013.

[3] B. B. Avants, C. L. Epstein, M. Grossman, and J. C. Gee. Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain. *Med Image Anal.*, 12(1):26–41, Feb. 2008.

[4] I. Bickle. Fibrolamellar hepatocellular carcinoma: case study, radiopaedia.org, Dec 2013. URL https://radiopaedia.org/cases/fibrolamellar-hepatocellular-carcinoma-1. [Online; Last accessed 21-April-2019].

[5] H. J. Bierens. The nadaraya watson kernel regression function estimator. In *Topics in Advanced Econometrics*, pages 212–247. Cambridge University Press, 1994.

[6] C. Chen and O. Öktem. Indirect image registration with large diffeomorphic deformations. *SIAM Journal on Imaging Sciences*, 11(1):575–617, 2018.

[7] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu. Spatial transformer networks. In *NIPS 28*, pages 2017–2025, 2015.

[8] C. Jud, N. Möri, B. Bitterli, and P. C. Cattin. Bilateral regularization in reproducing kernel hilbert spaces for discontinuity preserving image registration. In *MLMI 2016, Athens, Greece, Oct. 2016*, pages 10–17, 2016.

[9] C. Jud, N. Möri, and P. C. Cattin. Sparse kernel machines for discontinuous registration and nonstationary regularization. In *CVPR, Las Vegas, NV*, pages 449–456, June 2016.

[10] M. Li, W. Bi, J. T. Kwok, and B. Lu. Large-scale nyström kernel matrix approximation using randomized svd. *IEEE Trans. on Neural Networks and Learning Systems*, 26(1): 152–164, Jan 2015.

[11] I. B. Malone, D. Cash, G. R. Ridgway, D. G. Macmanus, S. Ourselin, N. C. Fox, and J. M. Schott. Miriad-public release of a multiple time point alzheimer's mr imaging dataset. *Neuroimage*, 70:33–36, Dec 2012.

[12] Akshay Pai, Stefan Sommer, Lauge Sørensen, Sune Darkner, Jon Sporring, and Mads Nielsen. Kernel bundle diffeomorphic image registration using stationary velocity fields and wendland basis functions. *IEEE Trans. Med. Imaging*, 35(6):1369–1380, 2016.

[13] J. Pluim, J. B. Maintz, and M. A. Viergever. Interpolation artefacts in mutual information-based image registration. *Computer Vision and Image Understanding*, 77 (2):211–232, Feb. 2000.

[14] C. E. Rasmussen and C. K. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, 2006.

[15] S. Remes, M. Heinonen, and S. Kaski. Non-stationary spectral kernels. In *NIPS 30*, pages 4642–4651, 2017.

[16] G. K. Rohde, A. Aldroubi, and B. M. Dawant. The adaptive bases algorithm for intensity-based nonrigid image registration. *IEEE Transactions on Medical Imaging*, 22(11):1470–1479, Nov 2003.

[17] G. K. Rohde, A. Aldroubi, and D. M. Healy. Interpolation artifacts in sub-pixel image registration. *IEEE Trans. on Image Processing*, 18(2), Feb. 2009.

[18] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D.J. Hawkes. Nonrigid registration using free-form deformations: Application to breast mr images. *IEEE Trans. on Medical Imaging*, 18(8):712–721, Aug. 1999.

[19] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, Cambridge, MA, 2001.

[20] L. A. Schwarz. *Non-rigid Registration Using Free-form Deformations, Diploma Thesis in Computer Science*. PhD thesis, Technical University at Munich, 2007.

[21] H. Soleimani and M.A. Khosravifard. Reducing interpolation artifacts for mutual information based image registration. *J Med Signals Sens*, 1(3):177–183, Jul. 2011.

[22] S. Sommer, F. Lauze, M. Nielsen, and X. Pennec. Sparse multi-scale diffeomorphic registration: the kernel bundle framework. *Journal of Mathematical Imaging and Vision, Springer Verlag*, 46(3):292–308, 2013.

[23] A. Sotiras, C. Davatzikos, and N. Paragios. Deformable medical image registration: A survey. *IEEE Trans. on Medical Imaging*, 32(7), July 2013.

[24] R. Sprengel, K. Rohr, and H. S. Stiehl. Thin-plate spline approximation for image registration. In *18th Annual Intl Conf. IEEE Engineering in Medicine and Biology Society*, Nov. 1996.

[25] A. G. Wilson and H. Nickisch. Kernel interpolation for scalable structured gaussian processes. In *ICML 2015*, volume 37, pages 1775–1784, July 2015.

[26] Z. Yaniv, B. C. Lowekamp, H. J. Johnson, and R. Beare. Simpleitk image-analysis notebooks: a collaborative environment for education and reproducible research. *J Digit Imaging.*, 31(3):290–303, 2018.