Contents lists available at ScienceDirect

Journal of Computational Mathematics and Data Science

journal homepage: www.elsevier.com/locate/jcmds

Reducing the Gibbs effect in multimodal medical imaging by the Fake Nodes approach

Davide Poggiali [a,b,*], Diego Cecchin [c], Stefano De Marchi [a,d]

[a] *PNC - Padova Neuroscience Center, University of Padova, Padua, Italy*
[b] *FAR Networks srl, Cernusco S.N., Milan, Italy*
[c] *Nuclear Medicine Unit, Department of Medicine - DIMED, Padua University Hospital, Padua, Italy*
[d] *Dipartimento di Matematica "Tullio Levi-Civita", University of Padova, Padua, Italy*

ARTICLE INFO

ABSTRACT

It is a common practice in multimodal medical imaging to undersample the anatomically-derived segmentation images to measure the mean activity of a co-acquired functional image. This practice avoids the resampling-related Gibbs effect that would occur in oversampling the functional image. As sides effect, waste of time and efforts are produced since the anatomical segmentation at full resolution is performed in many hours of computations or manual work. In this work we explain the commonly-used resampling methods and give errors bound in the cases of continuous and discontinuous signals. Then we propose a Fake Nodes scheme for image resampling designed to reduce the Gibbs effect when oversampling the functional image. This new approach is compared to the traditional counterpart in two significant experiments, both showing that Fake Nodes resampling gives smaller errors at the cost of an higher computational time.

1. Introduction

In nowadays medical imaging it is becoming more and more popular the usage of multimodal imaging. The typical setup of a multimodal imaging system allows to acquire simultaneously both anatomical and functional images of a physical body [1,2], as for instance SPECT/CT,[1] PET/MRI[2] or PET/CT. Anatomical imaging (e.g. CT, MRI) aims to picture accurately the interiors of the body and offers a high spatial resolution, around 1 mm^3 per voxel with the present day's machines. To the other side, functional imaging (SPECT, PET in our example) aims to show physiological activity of the body under examination, offering an high sensitivity at the price of a lower spatial resolution (about $8 - -27$ mm^3 per voxel) than the morphological counterpart.

In this context, in many clinical studies researchers need to measure some statistical moments (usually the mean or median) of the functional activity inside some specific segment previously identified from the anatomical image, see e.g. [3]. In order to get such measure it is necessary that the anatomical and functional image have the exact same shape, which ensures that the position of all the voxels of both images correspond. There are usually two ways to achieve that:

1. oversampling the functional image to the same size of the anatomical,
2. undersampling the anatomical image and the segments to the same size of the functional image [4,5].

In a recent paper [6] we showed that the latter is preferable due to the Gibbs effect that occurs in oversampling. However we found that this approach would be a waste of the time and efforts that are spent in producing accurate segments at full resolution [7]. For these reasons we have been looking for a Gibbs-free oversampling technique.

Inspired by the Fake Nodes Approach introduced in [8,9], in this work we propose a Fake Nodes-based interpolation technique that allows to oversample the functional image that takes into account the segments at full resolution, resulting in a remarkable reduction of the error due to the Gibbs effect. After a description of the interpolation scheme behind the Fake Nodes Approach, we discuss some theoretical results and some experiments.

The paper is organized as follows. After this Introduction, in Section 2 we recall the interpolation methods currently used in image resampling and give an error bound for these methods. In Section 3 we describe the Fake Nodes interpolation and the new scheme based on Fake Nodes in multimodal image resampling. In Section 4 we describe two significant experiments to compare a commonly used resampling method with the Fake Nodes-based approach. The first experiment is *in silico* using an analytically-defined test image. The second one is *in vitro*, using a set of PET/CT scans of a phantom for with known foreground to background ratio.

In the last section we conclude outlining some future research directions.

2. Interpolation methods for image resampling

2.1. Image definition

An image is given by the signal intensity at a given point in space. Mathematically we may define an **image function**

$$f : \Omega \subseteq \mathbb{R}^3 \longrightarrow \mathbb{R} \tag{1}$$

where the domain is a parallelepiped $\Omega = [a_1, b_1] \times [a_2, b_2] \times [a_3, b_3]$. In practice the information on an image function are the its values sampled over a regular and equispaced grid

$$X = \left\{ (\boldsymbol{x}_{ijk}) = (x_i, y_j, z_k) \in \Omega \right\} \quad \text{with} \quad \begin{matrix} i = 1, \dots, n_x \\ j = 1, \dots, n_y \\ k = 1, \dots, n_z \end{matrix} . \tag{2}$$

The values are stored in the matrix $F_{ijk} = f(\boldsymbol{x}_{ijk})$ which is indeed $f(X)$. From now on, F is referred simply as the **image**. For the sake of simplicity and without loss of generality we may consider the cube $\Omega = [a, b]^3$ equally-sampled along the axes, i.e. $n_x = n_y = n_z = n$, so that $x_i = y_i = z_i = \frac{b-a}{n-1}(i-1) + a$, $i = 1, \dots, n$.

2.2. Interpolation for image resampling

Resampling an image to another equispaced grid, say X', is equivalent to find $f(X')$, which is in general different from $f(X)$. This can be done by linear interpolation.

We shall concern with real-valued functions defined in Ω. In Ω we the consider a set $X = \{x_1, \dots, x_n\}$ of nodes, the so-called node set. For each x_i a value, say $y_i \in \mathbb{R}$, is given. The interpolation problem is finding a suitable function $p : \Omega \longrightarrow \mathbb{R}$ such that $p(x_i) = y_i$, $i = 1, \dots, N$. Usually p is chosen from some family of functions on Ω. In this work we are interested to interpolation from a finite dimensional linear space of functions.

Definition 2.1. Let f be a function from a N-dimensional vector space of functions on Ω. Let B this vector space with basis $\{b_1, \dots, b_N\}$. The **interpolant** of f is then the linear combination

$$\mathcal{P}f = \sum_{i=1}^{N} c_i b_i \tag{3}$$

with $c_i \in \mathbb{R}$ found by imposing the conditions

$$\mathcal{P}f_{|X} = f_{|X} . \tag{4}$$

The interpolant is unique provided that

$$\det \left[b_i(x_j) \right]_{i,j=1,\dots,N} \neq 0.$$

With the notation $\mathcal{P}f_{|X} = f_{|X}$ we mean $\mathcal{P}f(\bar{x}) = f(\bar{x}) \;\; \forall x \in X$.

In the case of image resampling, we choose the interpolant $\mathcal{P}f$ as the (possibly unique) linear combination of a chosen set of basis functions

$$\mathcal{B} = \{B_{ijk}\}_{i,j,k=1,\dots,n}$$

such that

$$\mathcal{P}f(\boldsymbol{x}_{ijk}) = f(\boldsymbol{x}_{ijk}) \;\; \forall i, j, k = 1, \dots, n.$$

To compute the resampled image it is sufficient to evaluate the interpolant over the evaluation grid $\mathcal{P}f(X')$ and store its value in a 3-dimensional array.

In image resampling, since the number of voxels n^3 can be extremely large, it is necessary to make some assumptions on the basis functions to compute the resampling image in a relatively short time. Accordingly with [6,10,11], the basis should satisfy the following characteristics.

(I) **Separable**: given a function b the basis B_{ijk} is then the product of translates of b at the samples, that is

$$B_{ijk}(x, y, z) := b(x - x_i)\, b(y - y_j)\, b(z - z_k), \tag{5}$$

$x \in \Omega, \quad i, j, k = 1, \dots, n.$

(II) **Cardinal**:

$$B_{ijk}(\boldsymbol{x}_{i'j'k'}) = \begin{cases} 1 & \text{if } (i, j, k) = (i', j', k') \\ 0 & \text{else} \end{cases} \tag{6}$$

(III) **Normalized**:

$$\sum_{ijk=1}^{n} B_{ijk}(\boldsymbol{x}) = 1 \quad \forall \boldsymbol{x} \in [a, b]^3. \tag{7}$$

(IV) **Compact support**:

$$\exists \alpha \in \mathbb{R}^+ \text{ s.t. } B_{ijk}(\boldsymbol{x}) = 0 \ \forall \|\boldsymbol{x} - \boldsymbol{x}_{ijk}\|_\infty \geq \alpha. \tag{8}$$

The reader can find some examples of basis functions satisfying all these characteristics in [6,10]. For instance the so-called *trilinear interpolation* corresponds to choose the function b as

$$b(x) = \begin{cases} 1 - \frac{|x|}{h} & \text{if } |x| \leq h \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

with $h = \frac{b-a}{n}$ the distance between two samples, and to define the basis functions B_{ijk} by consequence as in Eq. (5).

Under the hypothesis (I) we can write the interpolant function as

$$\mathcal{P}f(\boldsymbol{x}) = \sum_{ijk=1}^{n} f(\boldsymbol{x}_{ijk}) B_{ijk}(\boldsymbol{x}). \tag{10}$$

Such interpolant can be quickly computed as the result of three subsequent tensor products as the following holds for hypothesis (II)

$$\mathcal{P}f(x, y, z) = \sum_{k=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} F_{ijk} b(x - x_i)\, b(y - y_j)\, b(z - z_k) \tag{11}$$

or, in Einstein's summation convention

$$\mathcal{P}f(x, y, z) = F_{ijk} b(x - x_i)\, b(y - y_j)\, b(z - z_k).$$

It has to be noticed that such formulation of image resampling and the theoretical results presented in this work can be easily extended to any dimension.

In particular, it is possible to speed up the computation of the interpolant, as it is shown by the following Lemma, whose proof is given in [6].

Lemma 2.1. *If $\mathcal{P}f$ is an interpolant of f satisfying (I), (II) and (IV) then, for every evaluation point $(x, y, z) \in [x_p, x_{p+1}] \times [y_q, y_{q+1}] \times [z_r, z_{r+1}]$ it exists a strictly positive integer $a \in \mathbb{N}$ such that*

$$\mathcal{P}f(x, y, z) = \sum_{k=max(1, r-a)}^{min(n, r+a-1)} \sum_{j=max(1, q-a)}^{min(n, q+a-1)} \sum_{i=max(1, p-a)}^{min(n, p+a-1)} F_{ijk} b(x - x_i)\, b(y - y_j)\, b(z - z_k). \tag{12}$$

This means that the local compactness of the support of the basis functions ensures that in each dimension only the a nodes before and after the evaluation point have an active role in evaluating the interpolant.

2.3. Error bound for image resampling

In this section we give an error bound for the image interpolation which relies on the *modulus of continuity* of a function [12–15].

Definition 2.2. Let $f : \Omega \subseteq \mathbb{R}^d \longrightarrow \mathbb{R}$ be a piecewise continuous function. The function f admits **a modulus of continuity** $\omega_{\boldsymbol{x}_0}(\delta) : \mathbb{R}^+ \longrightarrow \mathbb{R}^+$ in $\boldsymbol{x}_0 \in \Omega$ if

$$|f(\boldsymbol{x}) - f(\boldsymbol{x}_0)| \leq \omega_{\boldsymbol{x}_0}(\delta) \ \forall \boldsymbol{x} \in \Omega \text{ s.t. } \|\boldsymbol{x} - \boldsymbol{x}_0\| \leq \delta.$$

where $\|\cdot\|$ is an arbitrary norm.

The definition of modulus of continuity given here is slightly relaxed from the one given in literature in order to include piecewise continuous functions. Namely, in our formulation it is not necessarily true that

$$\lim_{\delta \to 0} \omega_{x_0}(\delta) = 0.$$

We can extend this definition and consider the **global modulus of continuity**, say $\omega(\delta)$,

$$|f(x) - f(y)| \leq \omega(\delta) \ \forall x, y \in \Omega \text{ s.t. } \|x - y\| \leq \delta.$$

It is noteworthy to recall that if a function admits a local modulus of continuity for every $x_0 \in \Omega$, then admits a global modulus of continuity

$$\omega(\delta) := \sup_{x_0 \in \Omega} \omega_{x_0}(\delta).$$

We now enounce the definition of discontinuity jumps, that we will use for quantifying the discontinuities of a piecewise continuous signal.

Definition 2.3. Let $f : \Omega \subseteq \mathbb{R}^d \longrightarrow \mathbb{R}$ be piecewise continuous on pairwise disjoint subdomains $\Omega_1, \dots, \Omega_m$ such that

$$\Omega = \overset{\circ}{\bigcup_{i=1}^{m}} \Omega_i.$$

The **jump** between two subdomains Ω_i, Ω_j, $i \neq j$ is the positive real

$$D(\Omega_i, \Omega_j) := \sup \left(J_{ij} \bigcup J_{ji} \right)$$

with

$$J_{ij} = \left\{ \lim_{x \to y} |f(x) - f(y)| , \ x \in \Omega_i, y \in \partial_i \Omega_j \right\}$$

were $\partial_i \Omega_j := \partial \Omega_i \bigcap \Omega_j$ is the part of the border of Ω_i that belongs to Ω_j.

In the special case that $\partial \Omega_i \cap \partial \Omega_j = \emptyset$, and consequently J_{ij} and J_{ji} are empty sets, the jump is considered equal to zero.

We now enounce two useful Lemmas for the estimation of the error at a given evaluation point.

Lemma 2.2. *A d-variate, Lipschitz-continuous function f admits a modulus of continuity of the form*

$$\omega(\delta) = K\delta$$

being K the Lipschitz constant.

The proof follows trivially from the definitions of Lipschitz continuity and global modulus of continuity.

Lemma 2.3. *A piecewise d-variate, Lipschitz-continuous function f admits a modulus of continuity of the form*

$$\omega(\delta) = K\delta + D$$

Proof. Suppose that domain Ω is the union of m pairwise disjoint subdomains $\Omega_1, \dots, \Omega_m$,

$$\Omega = \overset{\circ}{\bigcup_{i=1}^{m}} \Omega_i$$

and in each Ω_i f is Lipschitz-continuous with constant k_i.

Considering two generic points $x, y \in \Omega$, we have to analyze three different cases.

Case 1: If $x, y \in \Omega_i$, the thesis follows by Lemma 2.2 with $K = k_i$ and $D = 0$.

Case 2: Suppose now that the points x and y belong to two neighboring subdomains, namely $x \in \Omega_i$ and $y \in \Omega_j$ with $\partial \Omega_i \cap \partial \Omega_j \neq \emptyset$. As in Definition 2.3, for any choice of $\xi_i \in \Omega_i$ and $\xi_j \in \partial_i \Omega_j = \partial \Omega_i \bigcap \Omega_j$ we get

$$|f(x) - f(y)| \leq |f(x) - f(\xi_i)| + |f(\xi_i) - f(\xi_j)| + |f(\xi_j) - f(y)|. \tag{13}$$

This inequality still true for the limit

$$|f(x) - f(y)| \leq \lim_{\xi_i \to \xi_j} \left\{ |f(x) - f(\xi_i)| + |f(\xi_i) - f(\xi_j)| + |f(\xi_j) - f(y)| \right\} \leq$$

$$\leq k_i \delta + D(\Omega_i, \Omega_j) + k_j \delta$$

for all $x, y \in \Omega$ such that $\|x - y\| \leq \delta$.

The thesis follows with $D = D(\Omega_i, \Omega_j)$ and $K = k_i + k_j$. This inequality also holds when $\partial_i \Omega_j$ is an empty set. In fact in such case $\partial_j \Omega_i$ cannot be empty as, so we can take $\xi_i \in \partial_j \Omega_i$, $\xi_j \in \Omega_j$ and the limit for $\xi_j \to \xi_i$ in (13).

Case 3: As last possible case we now suppose that the points x and y belong to two non-neighboring subdomains Ω_i and Ω_j respectively, with $\partial\Omega_i \cap \partial\Omega_j = \emptyset$. In this case the inequality (13) is repeated for all the subdomains intersecting the straight line between $x \in \Omega_i$ and $y \in \Omega_j$. In the worst case such line touches all the m subdomains. Hence the claim follows with

$$K = m \left(\max_{1 \le i \le m} k_i \right)$$

and

$$D = m \left(\max_{1 \le i,j \le m} D(\Omega_i, \Omega_j) \right). \quad \square$$

We now prove an upper bound for the pointwise interpolation error given by any modulus of continuity.

Theorem 2.4. *Let f be a trivariate and bounded function admitting a local modulus of continuity $\omega_x(\delta)$. Let $\mathcal{P}f$ be its interpolant built according to (I), (II), (III) and (IV). Then, there exits $\delta^* > 0$ such that*

$$|\mathcal{P}f(x) - f(x)| \le \omega_x(\delta^*).$$

Proof. From Eq. (10) and from (III) we get

$$\mathcal{P}f(x) - f(x) = \sum_{ijk=1}^{n} f(x_{ijk}) B_{ijk}(x) - f(x) \left(\sum_{ijk=1}^{n} B_{ijk}(x) \right)$$
$$= \sum_{ijk=1}^{n} \left(f(x_{ijk}) - f(x) \right) B_{ijk}(x).$$

We know from Lemma 2.1 that only some nodes in a neighborhood of x have to be taken into account in the interpolation formula. To be precise the x_{ijk} such that $\|x - x_{ijk}\|_\infty < \alpha$. Since f admits a modulus of continuity we can state

$$\left| f(x_{ijk}) - f(x) \right| \le \omega_x(\delta^*),$$

with $\delta^* = \alpha$ and i, j, k in the range indicated in Lemma 2.1. By using the normalized basis

$$\mathcal{P}f(x) - f(x) \le \sum_{ijk=1}^{n} \omega_x(\delta^*) B_{ijk}(x) = \omega_x(\delta^*).$$

and similarly

$$\mathcal{P}f(x) - f(x) \ge -\omega_x(\delta^*),$$

which proves the Theorem.

In a similar way we can find a global upper bound to the interpolation error, simply by taking the supremum of the local continuity moduli.

Corollary 2.4.1. *Let f be a trivariate and bounded function admitting a global modulus of continuity $\omega(\delta)$. Let $\mathcal{P}f$ be its interpolant built according to (I), (II), (III) and (IV). Then, there exits $\delta^* > 0$ such that*

$$|\mathcal{P}f(x) - f(x)| \le \omega(\delta^*).$$

2.4. Gibbs effect in image oversampling

We usually refer to Gibbs effect as the oscillations around a discontinuity that occurs when approximating a signal with a truncated Fourier series [16,17]. Such effect is present also in image (re)sampling, often called "ringing effect" for the wavelike, concentric oscillations that appears around any sudden change of signal intensity [18,19]. Unlike the *Runge effect*, the magnitude of the oscillation goes to a plateau as the number of interpolation nodes goes to infinity. The presence of the Gibbs effect in multimodal medical imaging has already been discussed in [6] where a "natural" error analysis has been presented. Similar results can be obtained by the results of the previous section. In fact, we can build the basis functions so that their support is $\alpha \propto \frac{1}{n}$. Hence, the interpolation error bound given in Theorem 2.4 goes to zero as $n \to \infty$ if the image function f is Lipschitz-continuous at the evaluation point x. By Lemma 2.2 indeed the image function admits a continuity modulus $\omega_x(\delta) \le K\delta$ that has the role of an error bound once evaluated on α, and goes to zero as $n \to \infty$. On the other hand, if f is only piecewise Lipschitz-continuous and the evaluation point belongs to the border of a subdomain Ω_i, then by Lemma 2.3 the image function admits a continuity modulus $\omega_x(\delta) \le K\delta + D$ that evaluated on α is an error bound, which converges to D as n increases to infinity.

In the next section we introduce the Fake Nodes interpolation, aiming to interpolating a discontinuous image on a mapped set of nodes so that it appears as continuous, getting rid of the Gibbs effect.

3. Fake Nodes interpolation

Fake Nodes is an interpolation paradigm introduced in [8,9] that can reduce the Gibbs effect in different frameworks and applications [20–22].

3.1. Definition of Fake Nodes interpolation

The Fake Nodes interpolant can be defined as in Definition 2.1, with the difference that the interpolation and evaluation nodes are subject to mapping. For reasons that will be clarified soon, in this paper we choose a slightly different definition of Fake Nodes with respect to the above quoted literature.

Definition 3.1. Let f be a d-variate function sampled on a set of n distinct nodes X. Let the set of functions $\{b_1, \ldots, b_m\}$, $m \geq n$ be a basis. Consider the map $S : \Theta \subseteq \mathbb{R}^d \to \mathbb{R}^d$, and $\mathcal{P}f$ the interpolant over $X^S \supseteq S(X)$ to $Y \supseteq f(X)$, X^S, Y both of cardinality m. We call **Fake Nodes interpolant** the function

$$\mathcal{R}^S f := \mathcal{P}f \circ S.$$

The following proposition holds.

Proposition 3.1. *The Fake Nodes interpolant as defined in Definition 3.1 satisfies the interpolation conditions (4), i.e* $\mathcal{R}^S f_{|X} = f_{|X}$.

Proof. Being $\mathcal{R}^S f = \mathcal{P}f \circ S$ and the fact that $\mathcal{P}f$ is the interpolant over the whole supersets X^S, Y, in particular we get

$$\mathcal{P}f(S(\bar{x})) = f(\bar{x}) \ \ \forall \bar{x} \in X,$$

that is

$$\mathcal{R}^S f(\bar{x}) = f(\bar{x}) \ \ \forall \bar{x} \in X. \ \ \square$$

Remark. This definition of Fake Nodes differs to classical ones since considers a larger number of basis functions and interpolation nodes. To use the image resampling defined in the previous section, we must have to consider a regular, equispaced grid as input. Since the mapped nodes $S(X)$ is not equispaced in general, we choose to interpolate over a larger grid by adding some nodes in order to obtain a regular and equispaced grid $X^S \supseteq S(X)$.

In correspondence of the newly added nodes, we can choose some values to form the set $Y \supseteq f(X)$. Since we want to avoid the Gibbs effect, which occurs next to a discontinuity, we shall choose such values in order to get a smooth, regular image. Given these choices, it becomes mandatory to use a larger basis of functions.

3.2. Fake Nodes approach for multimodal image resampling

As discussed in the Introduction, in multimodal imaging we have a high resolution morphological image I along with its segmentation M, composed by integer values indicating the number of segment each voxel belongs. We also have a low resolution functional image F, and we aim to estimate the mean value of the functional image inside each segment with the lowest possible error.

The image I is defined on a grid X_{high} of the domain $\Omega = [a, b]^3$. The segmentation is the set of pairwise disjoint subsets of the domain Ω (cf. e.g. [23]), that is

$$\Omega = \bigcup_{k=0}^{m} \Omega_k \quad \text{with} \quad \Omega_i \cap \Omega_j = \varnothing \ \ \forall i, j = 1, \ldots, m, \tag{14}$$

and is known up to the resolution of I, and stored in the matrix

$$M = \sum_{k=0}^{m} k \, \chi_{\Omega_k}(\boldsymbol{x}_{ijk}) \ \text{ with } \boldsymbol{x}_{ijk} \in X_{high} \tag{15}$$

of the same size of I. By convention, the zero-indexed segment represent the background.

On the other hand, the functional image F is defined on another, smaller grid of $[a, b]^3$, say X_{low}.

We then propose to oversample the functional image F to the dimension of I and M by using a Fake Nodes interpolation with mapping function

$$S(\boldsymbol{x}) = \boldsymbol{x} + \sum_{k=0}^{m} \alpha_k \chi_{\Omega_k}(\boldsymbol{x}), \tag{16}$$

where $\boldsymbol{\alpha}_k = ((b - a)k, 0, 0)$. In other words, we create as many images as the segments, each with the values of the functional image inside the relative segment and zero elsewhere. These images are then stacked over the first axis, creating a long, unique 3-dimensional image.

As stated in the remark, this would be not sufficient to overcome Gibbs effect, as the so obtained signal has still some sudden change of intensity. Since we have the freedom to choose the values on the new nodes, we apply Gaussian blurring on the image and re-impose the original intensity values on the set $S(X_{low})$. This process can be repeated a chosen number of times, say n_s, until a smooth and continuous-like image is obtained. Our preliminary experiments showed that increasing the value of n_s did not lead to significant improvements after $n_s = 3$. For this reason the experiments below, we chose $n_s = 3$. The result of this process can be seen in Fig. 2.

The so-obtained image is then oversampled to the same resolution of I and M, and then recomposed according to the mapping S defined in Eq (16), as indicated in Definition 3.1. This method is explained in better detail in the following Algorithm 3.1.

Algorithm 3.1. Fake Nodes image resampling

Inputs:
I the target, high-resolution image;
M the segmentation of I in $m + 1$ segments;
F the input, low-resolution image;
$\mathcal{G}$ a Gaussian function of fixed variance.
Execution:

    1. Downsample the segmentation image M to the resolution of F, call the result M_{low};

    2. **for** *segm* in 0:*m*:

        2.1. Impose the values of the current segment to the image at *segm* level and add the result to a list of images *tempF*:

$$tempF[segm] = F[M_{low} == segm]$$

        2.2. **repeat** n_s **times:**

            2.2.1. Gaussian smoothing by convolution:

$$tempF[segm] = tempF[segm] * \mathcal{G}$$

            2.2.2. Re-imposition: $tempF[segm] = F[M_{low} == segm]$

    3. Stack the images *tempF[]* along the first axis, call the result *fakeF*;

    4. Oversample the image *fakeF* to the resolution of I with a basis of choice, call the result *HRfakeF*;

    5. Initialize the result image *highF* as a matrix with the size of I;

    6. **for** *segm* in 0:*m*:

        6.1. Impose the values at each segment:

$$highF[M == segm] = HRfakeF[M == segm]$$

Output:
highF, the resampled version of F using Fake Nodes.

Remark. The whole algorithm has a computational cost of m times the computational cost of the corresponding resampling algorithm, where m is the number of segments. This means that if the number of segments is too large it would be better to reduce it by merging some segments with similar values at the boundaries.

4. Experiments and results

For testing the Fake Nodes resampling and compare it to the usual resampling methods, we carried on two experiments. The first one is made on the Shepp–Logan phantom [24], an analytically-defined image with constant intensity in each of its ellipsoidal segments. Since the exact value is known, in this case we can compute the resampling errors. An example of Shepp–Logan phantom is reported in Fig. 1.

The second test is from a real PET/CT dataset of scans of the same physical phantom. This phantom is a closed container made of plastic. Inside the containers six spheres of diameters $10, 13, 17, 22, 28$ and 37 mm are filled with the tracer of known activity (we call 'hot' or foreground the interior of the spheres), whereas the remaining part of the phantom (here called 'cold' or background) is filled with water and tracer of known activity.

In both experiments we compare the results obtained by oversampling the functional image using the trilinear interpolation basis (9), cfr. [10] (Trilinear from now on) to the trilinear interpolation used in combination with the Fake Nodes scheme explained in the previous paragraph (Fake-Trilinear) and the usually-adopted method of undersampling the anatomical sement image to match the functional image resolution (Downscaling). The resampling is performed in ANTs [25,26], a popular software suite in neuroimaging written on top of ITK [27]. The Fake Nodes scheme is implemented in a set of Python scripts. All the scripts written for this paper and some sample data can be found in the repository of this paper https://github.com/pog87/FakeResampling3D.

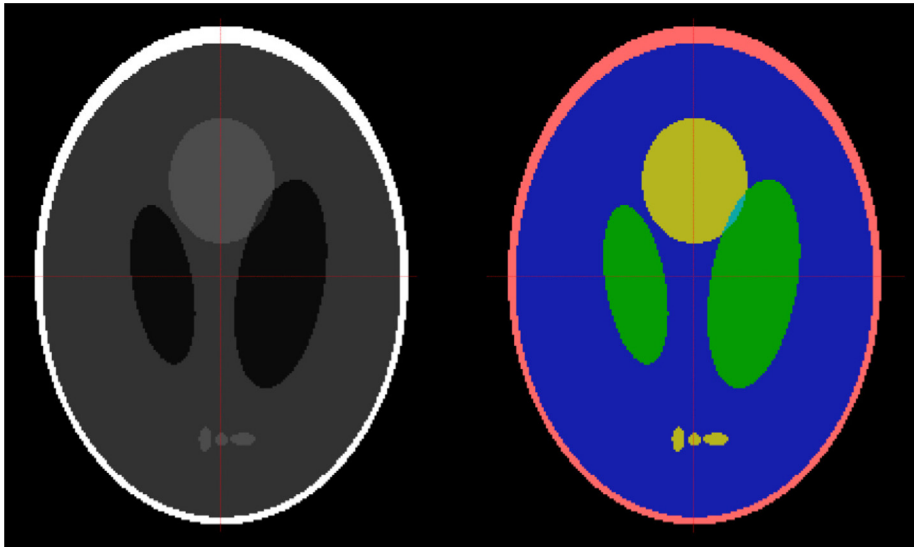Another package that can be useful to understand image resizing can be found at [28].

Fig. 1. Left: an axial slice of the Shepp–Logan phantom. Right: segmentation of the same slice, with the segments colored as follows: black segment number 0, red 1, green 2, blue 3, yellow 4, cyan 5. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Segment index and absolute error per segment (the smaller, the better) relative to the oversamplings of the Shepp–Logan phantom with Trilinear interpolation, Trilinear interpolation with Fake Nodes and Downscaling of the segment image. The reference values are indicated with different colors in the right parts of Fig. 1.

Segm. index	Trilinear	Fake-Trilinear	Downscaling
0	$2.0 \cdot 10^{-3}$	$9.9 \cdot 10^{-17}$	$6.7 \cdot 10^{-4}$
1	$1.8 \cdot 10^{-1}$	$9.3 \cdot 10^{-5}$	$1.3 \cdot 10^{-1}$
2	$6.9 \cdot 10^{-3}$	$1.4 \cdot 10^{-6}$	$4.6 \cdot 10^{-3}$
3	$1.6 \cdot 10^{-2}$	$2.4 \cdot 10^{-6}$	$1.2 \cdot 10^{-2}$
4	$3.9 \cdot 10^{-3}$	$7.9 \cdot 10^{-6}$	$2.4 \cdot 10^{-3}$
5	$1.0 \cdot 10^{-2}$	$6.1 \cdot 10^{-5}$	$2.7 \cdot 10^{-3}$

4.1. Experiment 1: Shepp-Logan phantom

We generated two Shepp–Logan phantom images: an high resolution image of size 256^3 and a low resolution image of 128^3 voxels with the Python packages `tomopy` and `nibabel` [29].

Since the Shepp–Logan phantom is piecewise constant, segmentation has been performed simply by identifying the voxels of the same values. The phantom and its segmentation can be observed in Fig. 1. The low resolution 'functional' image was resampled to the high resolution using Trilinear interpolation, Fake Nodes Trilinear interpolation, and the usually adopted method of downscaling the segment to the lower resolution. The Fake Image generated for resampling can be observed in Fig. 2. At last, the mean value per segment has been computed and compared with the actual, constant value of the segment.

As a second error measure we compare the errors between the global errors of the Trilinear and Fake-Trilinear oversampling with respect to the high resolution phantom. The error is evaluated using three different metrics: Root Mean Square Error (RMSE), Peak Signal to Noise Ratio (PSNR) [30] and inverse Structural Similarity Index Measure (1-SSIM) [31].

The results can be observed in Fig. 3 and Tables 1 and 2. As can be easily seen, the Fake Nodes resampling leads to more accurate mean value for every segment, including the background. The Fake-Trilinear oversampling also outperformed the Trilinear interpolation in all the chosen metrics.

4.2. Experiment 2: PET/CT phantom data

The RIDER PET/CT Phantom dataset [32] has been downloaded from the dataset official webpage hosted on the Cancer Imaging Archive (TCIA) [33] website. This dataset consists in 20 repeated scans of the RIDER PET/CT Phantom filled with Ge68 tracer and reconstructed using 3D Filtered Back Projection (FBP) [34]. We know that in principle the foreground-to-background ratio should be exactly equal to 4, as the interior of the spheres has been filled with a radioactive tracer of activity four times higher than the water lying outside them.
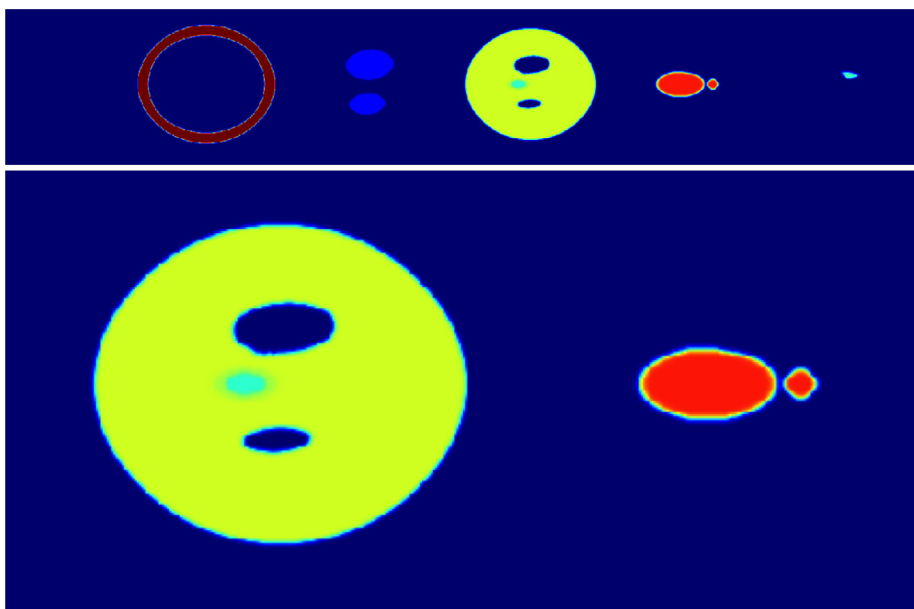
Fig. 2. Top: a Fake Image to be resampled, obtained from the low resolution image with Algorithm 3.1. Bottom: a zoom of the same, showing that the signal is smooth. The jet colormap (blue–green–yellow–red) has been used to better represent the smoothness of the signal. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 3. Left: a zoom of an axial slice of the high resolution Shepp–Logan phantom. Center: the same slice oversampled with Trilinear interpolation. Right: the same slice oversampled with Trilinear interpolation and Fake Nodes.

Table 2
Global errors of Trilinear and Fake-Trilinear oversampled images over the gold standard, high resolution image, in terms of Root Mean Square Error (RMSE, the smaller, the better), Peak Signal to Noise Ratio (PNSR, the closer to 100, the better) and inverse Structural Similarity Index Measure (1-SSIM, the smaller, the better).

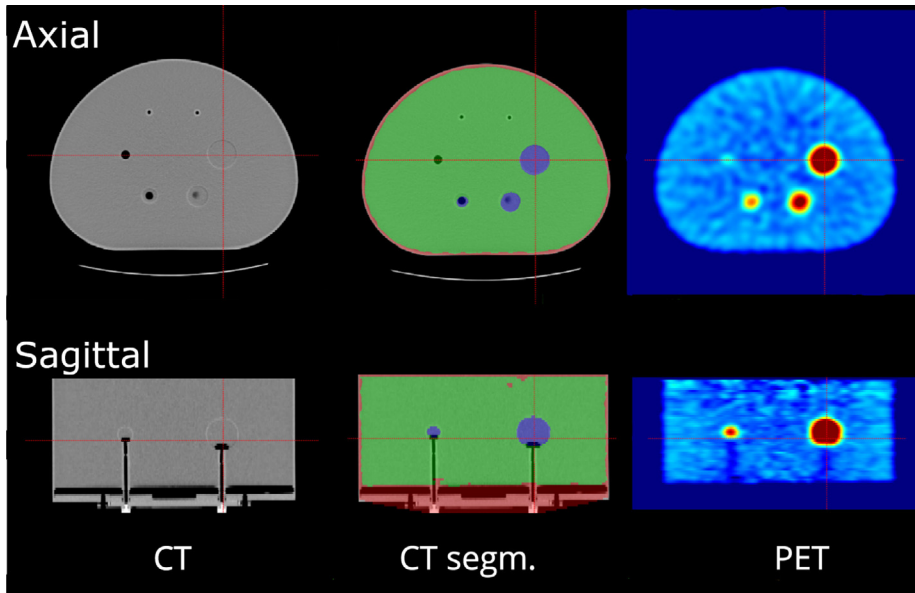|	RMSE	PNSR (dB)	1-SSIM
Trilinear error	$7.5 \cdot 10^{-2}$	22.5	$4.3 \cdot 10^{-2}$
Fake-Trilinear error	$8.5 \cdot 10^{-5}$	81.4	$5.4 \cdot 10^{-8}$

Fig. 4. The same axial (top) and sagittal (bottom) slices of the PET RIDER phantom CT (left), CT segmentation (center) and PET scan (right). The segmentation indexes are: background (black) 0, phantom shell (red) 1, 'cold' water (green) 2, 'hot' spheres (violet) 3. The jet colormap has been used for PET images. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
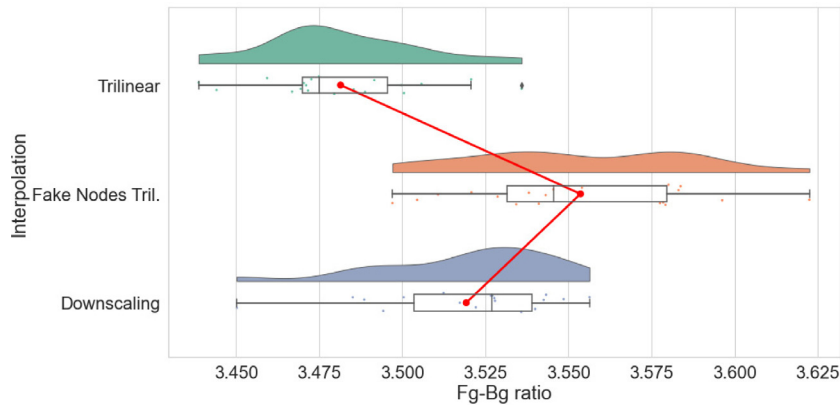

Fig. 5. Results of the RIDER phantom experiments, pictured as Raincloud plots. Comparison between foreground to background ratios obtained using Trilinear interpolation (top, green), Fake Nodes Trilinear interpolation (middle, orange) and Downscaling of the CT segment image to the PET resolution (bottom, purple). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The CT scan has been automatically segmented using k-means clustering in four segments: exterior, phantom plastic shell, background 'cold' water and foreground 'hot' spheres. A slice of a phantom scan can be found in Fig. 4. The foreground to background ratio of each scan has been computed as the average of PET signal in the foreground segment over its average in the foreground segment

$$FBr = \frac{avg(PET[foreground])}{avg(PET[background])} \tag{17}$$

for each of the three interpolation methods described above. In all of the cases a Partial Volume Correction (PVC) method [35] was applied to resampled PET before computing the mean value per segment, with an estimated Full Width at Half Maximum (FWHM) [36] of 6 mm. Even if PVC still does not completely exclude the possibility errors in the computation on FBr, this should not affect the difference between the interpolation methods.

The results of this experiment can be observed in the Raincloud plot [37] at Fig. 5. The Fake Nodes approach results in a mean error reduced of about 14% with respect to the usual Trilinear resampling and 6% lower than the error produced by the undersampling method. The t-test between the FBr of Fake-Trilinear groups gives a *p*-value of about $p \approx 0.7 \cdot 10^{-8}$ with respect to Trilinear and $p \approx 0.003$ with respect to Undersampling, indicating that the Fake Nodes approach leads to a significantly better result.

4.3. Comments to the results

The first experiment shows that resampling a functional image with the Fake Nodes approach results in an almost perfect set of mean values for each segment, with an impressive improvement with respect both to the Trilinear image resampling and to the Undersampling method. In a less controlled and more realistic situation, the use of Fake Nodes in resampling does lead to a significant improvement in terms of FBr, but still the result appears not close enough to the target value of 4. These results are coherent with the dataset documentation [32], indicating that an exact quantification of the tracer concentration in PET is still far from being achieved.

5. Conclusions and future work

In this paper we have proposed a Fake Nodes scheme for medical image oversampling that can be applied to any image of arbitrary dimension. This scheme allows to achieve an higher accuracy at the cost of a larger computational time. In Sections 2 and 3, we introduced image resampling techniques as interpolatory methods and gave an error bound which is different in the cases of continuous and piecewise continuous signals. Fake Nodes approach has proved to bring a more accurate functional quantification in the experiments, both using an analytically-defined phantom and real data from the scans of a PET/CT physical phantom.

The results on the physical phantom appeared to be still too far from the target value of 4. This may indicate that the Partial Volume Correction, Attenuation Correction and a Fake Nodes-like oversampling should be all applied during the reconstruction phase in order to achieve an exact quantification in PET imaging.

The usage of Fake Nodes resampling could also in principle bring benefits in a generic image analysis context, provided that an high resolution segmentation is available.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

References

[1] Ehman EC, Johnson GB, Villanueva-Meyer JE, Cha S, Leynes AP, Larson PEZ, et al. PET/MRI: where might it replace PET/CT? J Magn Reson Imaging 2017;46(5):1247–62. http://dx.doi.org/10.1002/jmri.25711.

[2] Zhang XY, Yang ZL, Lu GM, Yang GF, Zhang LJ. PET/MR imaging: New frontier in Alzheimer's disease and other dementias. Front Mol Neurosci 2017;10:343. http://dx.doi.org/10.3389/fnmol.2017.00343.

[3] Cecchin D, Barthel H, Poggiali D, Cagnin A, Tiepolt S, Zucchetta P, et al. A new integrated dual time-point amyloid PET/MRI data analysis method. Eur J Nucl Med Mol Imaging 2017;44(12):2060–72. http://dx.doi.org/10.1007/s00259-017-3750-0.

[4] Tustison NJ, Avants BB, Cook PA, Kim J, Whyte J, Gee JC, et al. Logical circularity in voxel-based analysis: Normalization strategy may induce statistical bias. Hum Brain Mapp 2012;35(3):745–59. http://dx.doi.org/10.1002/hbm.22211.

[5] Dumitrescu, Boiangiu. A study of image upsampling and downsampling filters. Computers 2019;8(2):30. http://dx.doi.org/10.3390/computers8020030.

[6] Poggiali D, Cecchin D, Campi C, De Marchi S. Oversampling errors in multimodal medical imaging are due to the gibbs effect. Mathematics 2021;9(12). http://dx.doi.org/10.3390/math9121348.

[7] Delgado J, Moure JC, Vives-Gilabert Y, Delfino M, Espinosa A, Gómez-Ansón B. Improving the execution performance of FreeSurfer. Neuroinformatics 2014;12(3):413–21. http://dx.doi.org/10.1007/s12021-013-9214-1.

[8] De Marchi S, Marchetti F, Perracchione E, Poggiali D. Polynomial interpolation via mapped bases without resampling. J Comput Appl Math 2020;364:112347. http://dx.doi.org/10.1016/j.cam.2019.112347.

[9] De Marchi S, Marchetti F, Perracchione E, Poggiali D. Multivariate approximation at fake nodes. Appl Math Comput 2021;391:125628. http://dx.doi.org/10.1016/j.amc.2020.125628.

[10] Burger W, Burge MJ. Principles of digital image processing: core algorithms. Germany: Springer, Berlin/Heidelberg; 2009, p. 210–33.

[11] Getreuer P. Linear methods for image interpolation. Image Process Line 2011;1:238–59. http://dx.doi.org/10.5201/ipol.2011.g_lmii.

[12] Bugajewski D, Gulgowski J, Kasprzak P. On continuity and compactness of some nonlinear operators in the spaces of functions of bounded variation. Ann Mat Pura Appl (1923 -) 2015;195(5):1513–30. http://dx.doi.org/10.1007/s10231-015-0526-7.

[13] Aronszajn N, Panitchpakdi P. Extension of uniformly continuous transformations and hyperconvex metric spaces. Pacific J Math 1956;6(3):405–39. http://dx.doi.org/10.2140/pjm.1956.6.405.

[14] Costarelli D, Vinti G. Quantitative estimates involving K-functionals for neural network-type operators. Appl Anal 2018;98(15):2639–47. http://dx.doi.org/10.1080/00036811.2018.1466277.

[15] Costarelli D, Vinti G. A quantitative estimate for the sampling kantorovich series in terms of the modulus of continuity in orlicz spaces. Constr Math Anal 2019;8–14. http://dx.doi.org/10.33205/cma.484500.

[16] Jerri A. The gibbs phenomenon in Fourier analysis, splines, and wavelet approximations. Z Angew Math Mech 2005;85(3):224. http://dx.doi.org/10.1002/zamm.200590016.

[17] Fornberg B, Flyer N. The gibbs phenomenon for radial basis functions. In: The gibbs phenomenon in various representations and applications. Sampling Publishing; 2006.

[18] Lehmann T, Gonner C, Spitzer K. Survey: Interpolation methods in medical image processing. IEEE Trans Med Imaging 1999;18(11):1049–75. http://dx.doi.org/10.1109/42.816070.

[19] Chhoa JF. An adaptive approach to gibbs' phenomenon. (Master's Thesis), Hattiesburg, Mississippi: The University of Southern Mississippi; 2020.

[20] De Marchi S, Erb W, Francomano E, Marchetti F, Perracchione E, Poggiali D. Fake nodes approximation for magnetic particle imaging. In: 2020 IEEE 20th mediterranean electrotechnical conference (MELECON) Palermo, Italy, 16–18 June 2020. IEEE; 2020, p. 434–8. http://dx.doi.org/10.1109/melecon48756.2020.9140583.

[21] De Marchi S, Elefante G, Marchetti F. Stable discontinuous mapped bases: the gibbs–runge-avoiding stable polynomial approximation (GRASPA) method. Comput Appl Math 2021;40(8). http://dx.doi.org/10.1007/s40314-021-01688-z.

[22] Marchi SD, Elefante G, Marchetti F. On (β, γ)-Chebyshev functions and points of the interval. J Approx Theory 2021;271:105634. http://dx.doi.org/10.1016/j.jat.2021.105634.

[23] Pham DL, Xu C, Prince JL. Current methods in medical image segmentation. Annu Rev Biomed Eng 2000;2(1):315–37. http://dx.doi.org/10.1146/annurev.bioeng.2.1.315.

[24] Shepp L, Logan B. The Fourier reconstruction of a head section. IEEE Trans Nucl Sci 1974;21(3):21–43. http://dx.doi.org/10.1109/tns.1974.6499235.

[25] Avants B, Epstein C, Grossman M, Gee J. Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain. Med Image Anal 2008;12(1):26–41. http://dx.doi.org/10.1016/j.media.2007.06.004.

[26] Avants BB, Tustison NJ, Song G, Cook PA, Klein A, Gee JC. A reproducible evaluation of ANTs similarity metric performance in brain image registration. Neuroimage 2011;54(3):2033–44. http://dx.doi.org/10.1016/j.neuroimage.2010.09.025.

[27] Mccormick M, Liu X, Jomier J, Marion C, Ibanez L. ITK: enabling reproducible research and open science. Front Neuroinformatics 2014;8. http://dx.doi.org/10.3389/fninf.2014.00013.

[28] Shocher A. Resizeright. 2018, GitHub Repository. GitHub, https://github.com/assafshocher/ResizeRight.

[29] Brett M, Markiewicz CJ, Hanke M, Côté M-A, Cipollini B, McCarthy P, et al. nipy/nibabel: 3.2.1. Zenodo 2020. http://dx.doi.org/10.5281/ZENODO.4295521, URL https://zenodo.org/record/4295521.

[30] Chervyakov N, Lyakhov P, Nagornov N. Analysis of the quantization noise in discrete wavelet transform filters for 3D medical imaging. Appl Sci 2020;10(4):1223. http://dx.doi.org/10.3390/app10041223.

[31] Marchetti F. Convergence rate in terms of the continuous SSIM (cSSIM) index in RBF interpolation. Dolomites Res Notes Approx 2021;14(1).

[32] Muzi P, Wanner M, Kinahan P. Data from RIDER PHANTOM PET-CT. The Cancer Imaging Archive; 2015, http://dx.doi.org/10.7937/K9/TCIA.2015.8WG2KN4W, URL https://wiki.cancerimagingarchive.net/x/bIRXAQ.

[33] Clark K, Vendt B, Smith K, Freymann J, Kirby J, Koppel P, et al. The cancer imaging archive (TCIA): Maintaining and operating a public information repository. J Digit Imaging 2013;26(6):1045–57. http://dx.doi.org/10.1007/s10278-013-9622-7.

[34] Jha AK, Purandare NC, Shah S, Agrawal A, Puranik AD, Rangarajan V. PET reconstruction artifact can be minimized by using sinogram correction and filtered back-projection technique. Indian J Radiol Imaging 2014;24(02):103–6. http://dx.doi.org/10.4103/0971-3026.134379.

[35] Müller-Gärtner HW, Links JM, Prince JL, Bryan RN, McVeigh E, Leal JP, et al. Measurement of radiotracer concentration in brain gray matter using positron emission tomography: Mri-based correction for partial volume effects. J Cereb Blood Flow Metab 1992;12(4):571–83. http://dx.doi.org/10.1038/jcbfm.1992.81.

[36] Cecchin D, Poggiali D, Riccardi L, Turco P, Bui F, De Marchi S. Analytical and experimental FWHM of a gamma camera: Theoretical and practical issues. PeerJ 2015;3(2):e722. http://dx.doi.org/10.7717/peerj.722.

[37] Allen M, Poggiali D, Whitaker K, Marshall TR, Kievit RA. Raincloud plots: a multi-platform tool for robust data visualization. Wellcome Open Res 2019;4:63. http://dx.doi.org/10.12688/wellcomeopenres.15191.1.