Network Modeling Analysis in Health Informatics and Bioinformatics (2025) 14:38
https://doi.org/10.1007/s13721-025-00534-0

ORIGINAL ARTICLE

New 3D segmentation algorithm for modeling of kidney in positron emission tomography images

Oona Rainio[1] · Aino Latva-Rasku[1] · Jussi Hirvonen[1,2] · Juhani Knuuti[1] · Riku Klén[1]

Received: 7 February 2025 / Revised: 25 April 2025 / Accepted: 29 April 2025

Abstract

Background ^{15}O-water positron emission tomography (PET) imaging enables noninvasive quantification of renal blood flow. While there are several existing methods automatically locating the volume of interest (VOI) for a kidney, separate cortex and medulla VOIs are needed for PET modeling due their functional differences. To assist in this, we introduce a new three-dimensional segmentation algorithm for kidney.

Materials and methods Our algorithm chooses an initial kidney VOI, finds its longitudinal axis and the direction of the non-convex part, and then creates cortex and medulla VOIs and removes the renal vessel area from them. We evaluated the algorithm by using it to define cortex and medulla VOIs of left and right kidneys in dynamic total-body ^{15}O-water PET images of 35 human patients. For all the 70 kidneys, we plotted the cross-section of the VOIs from three different anatomic directions and asked two expert physicians to assess their quality. Additionally, we computed cortical and medullary renal blood flow estimates by fitting a compartment model to the mean time-activity curves of our VOIs.

Results According to the evaluation by the physicians, the cortex and medulla VOIs were mostly correct in all three directions for 78.6% of the total 70 kidneys and correct in at least one direction for 94.3% of the kidneys. The segmentation inaccuracies were typically caused by the algorithm placing cortex VOI partially outside of the kidney or in the medulla. However, regardless of these inaccuracies, all the VOIs were accurate enough to be used for compartment modeling. The resulting cortical and medullary blood flow were very close to the values reported in earlier studies with similar patient populations.

Conclusion Our proposed algorithm can be used to create an automatic pipeline for accurate quantification of cortical and medullary tracer perfusion after a PET scan.

Keywords Automatic segmentation · Compartmental modeling · Kidney · Positron emission tomography · Renal blood flow

1 Introduction

Kidneys are an important part of the human body, filtering blood in order to remove impurities and excess fluid and control the balance of numerous different substances. Impaired renal function is associated with high blood pressure (Wright-Nunes et al. 2012), diabetes (Thomas et al. 2015), and several cardiovascular diseases (Elsayed et al. 2007). Consequently, renal blood flow quantification offers useful information about the patient's health.

Due to the recent emergence of the new total-body imaging devices (Knuuti et al. 2023), the application possibilities of positron emission tomography (PET) have increased for the study of kidney as well as other organs. Based on recording how short-lived radioactive tracer substances, such as ^{15}O-labeled water or ^{18}F-fluorodeoxyglucose, accumulate in

✉ Oona Rainio
ormrai@utu.fi

Aino Latva-Rasku
aehyyp@utu.fi

Jussi Hirvonen
jussi.hirvonen@utu.fi

Juhani Knuuti
juhani.knuuti@varha.fi

Riku Klén
riku.klen@utu.fi

[1] Turku PET Centre, University of Turku and Turku University Hospital, Turku, Finland

[2] Department of Radiology, University of Turku and Turku University Hospital, Turku, Finland

Published online: 31 May 2025

different tissues in the human body after they have been injected into the patient's bloodstream, PET can offer precise information on body function and metabolism (Townsend et al. 2004). In particular, dynamic PET imaging is of interest as it gives us a way to view how the changes in the tracer distribution in the body occur with respect to time (Knuuti et al. 2023).

Compartment modeling is commonly utilized in the analysis of dynamic PET images in order to summarize the extensive amount of image data into a few relevant key parameters (Rainio and Klén 2025). This method relies on the assumption that the tracer is divided into a certain number of physiologically homogeneous compartments and the tracer exchange between them can be described with a set of differential equations. As each voxel in the three-dimensional (3D) image space of a dynamic PET image has a time-activity curve (TAC), we can fit either several voxel-wise separate compartment models to produce a parametric image of the modeling results or a single compartment model to the mean TAC of a specific volume of interest (VOI). The location of the desired VOIs is often a simple tasks given that, during the past decade, numerous convolutional neural networks and other machine learning (ML)-based automatic tools have been developed for automatic segmentation of different organs and tissue types in magnetic resonance imaging (MRI) (Arif et al. 2020; Kurata et al. 2019; Pereira et al. 2016) or computed tomography (CT) images (Liu et al. 2020; Osadebey et al. 2021; Park et al. 2019), and at least one of these two modalities is typically recorded during the PET imaging by the standard PET/MRI and PET/CT hybrid scanners.

However, a kidney is not homogeneous enough be used as a single compartment due to the high differences in the tracer concentration between the highly vascularized solid cortex and the fluid-filled medulla. As the functional differences inside the kidney are not similarly visible in the MRI and CT images, most of the automatic tools are only able to find a single VOI for the whole kidney (Daniel et al. 2021; da Cruz et al. 2020; Chen et al. 2022; Haghighi et al. 2018; Guo et al. 2022) or renal tumors (Yu et al. 2019). Additionally, this VOI might contain renal arteries and veins which causes overestimation of the perfusion of the kidney in the compartment model due the high tracer concentration in the arterial blood. Furthermore, since the MRI or the CT imaging is typically performed once before or after the dynamic PET imaging, any movement between the two imaging scans or during the dynamic PET imaging produces error to the location of the MRI- or CT-based VOIs with respect to the PET image.

Consequently, to address this issue, we introduce a new algorithm that first locates a robust kidney VOI from a PET image, then separates the cortex and the medulla within this VOI while also adjusting its initial outlines to more accurate locations, and finally removes the renal vessel area from the resulting segments. If used alone, this algorithm offers a semi-automatic method that requires the coordinates of a box containing the kidney in the PET image from the user or, if combined with TotalSegmentator or any other similar robust segmentation tool, it can be used to create a fully automatic pipeline for compartment modeling of cortical and medullary PET tracer perfusion. Additionally, the functions for subtasks in this algorithm might be of use for study of also other organs in dynamic or static PET images.

2 Materials and methods

2.1 Software

The algorithm was coded with Python (version: 3.9.9) (Rossum and Drake 2009) with SciPy (version: 1.7.3) (Virtanen et al. 2020), Rasterio (version: 1.3.11) (Gillies et al. 2013), and Shapely (version: 1.8.0) (Gillies et al. 2007). The image data and different VOI segmentations were visually checked with Carimas (version: 2.10) (Rainio et al. 2023). The automatic segmentation tool TotalSegmentator (version: 1.0) (Wasserthal et al. 2023) was used to locate kidney VOIs from CT images. Additionally, R (version: 3.4.1) (R Core Team et al. 2013) was used for model fitting and statistical testing.

2.2 Data and pre-processing

The algorithm was designed and evaluated by using data of 37 randomly selected patients who were referred for total-body ^{15}O-water PET perfusion imaging at Turku PET Centre in Turku, Finland, between August 2022 and January 2023 due to signs of a potential coronary artery disease. All the patients had two kidneys. During the imaging, the patients lay at rest in a supine position on a bed while a CT scan followed by a 280-seconds-long dynamic PET scan was taken with Biograph Vision Quadra (Siemens Healthineers) PET/CT scanner after an injection of ^{15}O-water dose with activity of 295–408 MBq. Additionally, plasma creatinine was measured to calculate glomerular filtration rate (GFR) with the chronic kidney disease-Epidemiology Collaboration (known as CKD-EPI) formula (Florkowski and Chew-Harris 2011).

The resulting PET images consisted of $220 \times 220 \times 380 \times 24$ image points with a voxel size of $1.65 \times 1.65 \times 2.80$ mm^3 and time intervals of 14·5 s, 3·10 s, 3·20 s, and 4·30 s. They were corrected for decay. The CT images had $512 \times 512 \times 380$ voxels with a voxel size of $0.977 \times 0.977 \times 2.80$ mm^3.

The CT images were given to TotalSegmentator to automatically obtain VOIs for left and right kidneys, after which both the CT images and their TotalSegmentator masks were

cropped and scaled to the size of the 3D frames of the PET images. By visually inspecting the TotalSegmentator kidney VOIs, it was noted that while they were correctly placed in the CT images, there was some movement between the PET and CT images, causing the misalignment of these VOI with respect to the PET data. Figure 1 shows this difference between CT (Fig. 1(A)) and PET images (Fig. 1(B)).

2.3 Algorithm

The PET and CT images and TotalSegmentator mask of the two patients were used to design the algorithm. The algorithm performs the following seven subtasks, also summarized in Fig. 2. If there is such a 3D kidney mask available that is approximately correct with respect to the PET images, be it manually drawn by a physician or automatically created with some kidney segmentation tool, the first two steps of this algorithm are unnecessarily. Additionally, if no potentially misaligned kidney mask from TotalSegmentator or another CT- or MRI-based segmentation tool is available, the rest of the steps after the first one still work as long as the user specifies the needed bounding box location.

2.3.1 Finding a minimum bounding box

Despite the misalignment of the TotalSegmentator kidney VOIs, the distance from these VOIs to the actual kidney locations in the PET data is not very great, so we use these VOIs to find the minimum bounding box approximately containing the kidney to be used as the bounding box of the new segmentation masks from the PET images. For either kidney, we consider the TotalSegmentator output showing only this kidney as a binary matrix and then choose the greatest connected component from the positive voxels with the help of *scipy.ndimage.measurements.label* in order to get rid of errors caused by a false positive segmentation of a few voxels fully outside the kidney. Then we create another binary mask with only this greatest connected component coded as positive and search the range of the coronal, sagittal, and transaxial slices with a non-zero maximum from this binary mask. Consequently, we obtain the index tuple $[k_1{:}k_2,k_3{:}k_4,k_5{:}k_6]$ that can be used to crop the minimum bounding box from the PET image approximately containing the kidney and no other organs.

2.3.2 Initial kidney mask

We estimated that about 30% of the area of the mean coronal or the mean sagittal slices inside the found bounding box $[k_1{:}k_2,k_3{:}k_4,k_5{:}k_6]$ belonged to kidney, so we create a binary mask showing the voxels whose mean value over time is at least 81st quantile of all the voxels within this box. As the kidney cortex has very high activity, this gives us a very accurate mask of the outlines of the kidney. To obtain a simply-connected VOI without holes in the middle despite the lower activity concentration in the medulla, we also include all such connected components formed out of the voxels below the 81st quantile that did not touch the sides of our bounding box. The resulting binary mask is considered initial version of the kidney mask. While it has certain inaccuracies, as shown in Fig. 1(C), it has higher overlap of the actual kidney than the original TotalSegmentator mask.

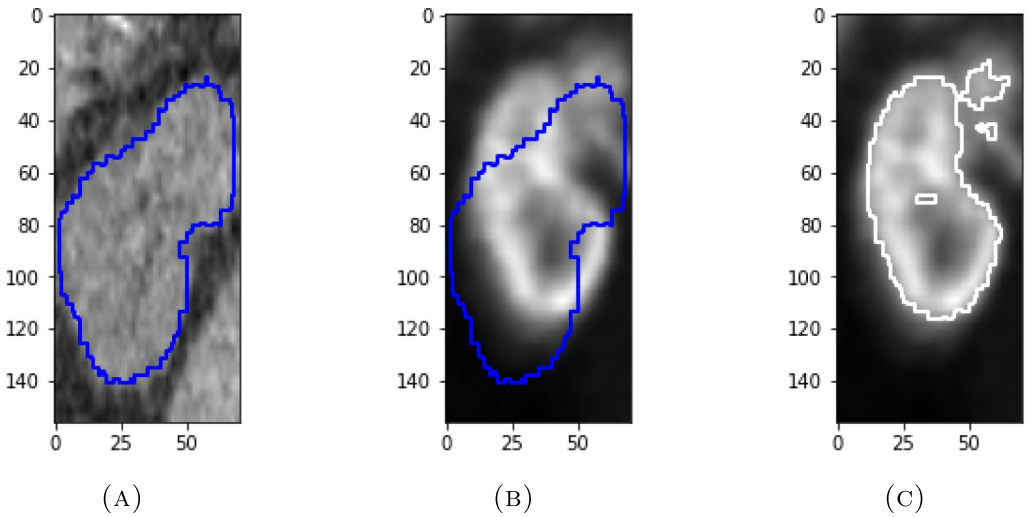


Fig. 1 A cropped coronal slice in the (**A**) CT image and (**B**) mean PET image over time showing the TotalSegmentator VOI for left kidney (outlined in blue) and (**C**) the same coronal slice of the mean PET image showing the robust kidney VOI (outlined in white) found with the help of a simple thresholding technique. The units of the axes express distance in millimeters

Kidney algorithm
findRenalMedullaAndCortex:
input: Dynamic total-body PET image **img4d** and binary TotalSegmentator mask **tsArr** of the initial misaligned (either left or right) kidney VOI;
output: the binary masks **medullaMask** and **cortexMask** of same size as a single time-frame of **img4d**

1. step: Finding a minimum bounding box
From the input **tsArr**, ***findBoundingBox*** returns the indexes $k_1, k_2, k_3, k_4, k_5, k_6$ of the bounding box.

2. step: Initial kidney mask
From the input **img4d**, $k_1, k_2, k_3, k_4, k_5, k_6$; ***findIniArr*** returns an initial kidney array **arr** of the size $(k_2 - k_1) \times (k_4 - k_3) \times (k_6 - k_5)$.

3. step: Locating the longitudinal kidney axis
scipy.measurements.center_of_mass gives **com** of **arr**. From the input **arr**, **com**; ***findTheAxis*** returns the angles θ, ϕ and the end points l_0, l_1 of the axis.

4. step: Locating the non-convex direction
From the input **arr**, **com**, l_0, l_1, θ, ϕ; ***findTheNonConvexDirection*** returns the angle μ and the direction point d of the non-convex direction.

5. step: Initial cortex mask
From the input **arr**, **com**,$l_0, l_1, \theta, \phi, \mu$; ***findAuxiliaryPoints*** returns the points x, y. From the input x, y, θ, ϕ, **arr**,**com**,**img4d**, $k_1, k_2, k_3, k_4, k_5, k_6$; ***findInitialCortexArr*** returns the initial cortex mask **iniArr** of same size as **arr**. With the input **iniArr**, l_0, l_1; ***connectCortexPoints*** adjusts **iniArr**.

6. step: Medulla and cortex separation
From the input **iniArr**, l_0, l_1; ***divideMedullaAndCortex*** returns **medullaArr** and **cortexArr**, both of same size as **arr**.

7. step: Removal of renal vessel area
From the inputs **medullaArr**, **com**, d, x, y and **cortexArr**, **com**, d, x, y; ***removeVesselArea*** returns an adjusted version of **medullaArr** and **cortexArr**. They are used to create **medullaMask** and **cortexMask**.

Fig. 2 The subtasks of our algorithm summarized

2.3.3 Locating the longitudinal kidney axis

The third step is to find the kidney's longitudinal axis passing though the kidney approximately parallel to its long sides. Namely, as the location and the angle of the kidney greatly varies between the different patients, this axis tells us how a kidney of a specific patient is actually positioned with respect to the 3D space of the PET image. While perhaps an intuitive approach would be search the maximal diameter of the kidney VOI, this will not work because the resulting axis would be diagonal with respect to the height of the kidney. Instead, we define here that a longitudinal axis is the line passing through the kidney so that it gives the smallest possible cross-section drawn at

the center of mass (COM) of the kidney perpendicular to this line.

The COM can be computed from our initial kidney mask with *scipy.ndimage.measurements.center_of_mass* and we can find all the possible lines passing through this point by spherical coordinates. Namely, each combination of the two angles $\theta \in [0, \pi/2]$ and $\phi \in [0, 2\pi)$ can be used to define a unique line passing through the COM and the point

$$q = \text{COM} + (h, \theta, \phi), \tag{2.1}$$

where $h \neq 0$ and

$$(h, \theta, \phi) = h[\sin(\theta)\cos(\phi), \sin(\theta)\sin(\phi), \cos(\theta)].$$

As can be seen from Fig. 3(A), the following three points

$$\begin{aligned} s_1 &= \text{COM} + (h, \pi/2, \phi + \pi/2), \\ s_2 &= \text{COM} + (h, \pi/2 - \theta, \phi + \pi), \\ s_3 &= \text{COM} + (h, \pi/2 + \theta, \phi), \end{aligned} \tag{2.2}$$

defined with any $h \neq 0$, form a straight angle on the plane orthogonal to this axis drawn at COM. However, in our code, we actually need to define the spherical coordinates as

$$(h, \theta, \phi) = h[\sin(\theta)\cos(\phi)/c_1, \sin(\theta)\sin(\phi)/c_2, \cos(\theta)/c_3] \tag{2.3}$$

so that the resulting points s_1, s_2, and s_3 correspond with the image points in the PET image with the voxel size of $c_1 \times c_2 \times c_3$ mm^3. In our code, $c_1 = 1.65$, $c_2 = 1.65$, and $c_3 = 2.80$. Also, we fix here

$$h = \max\{c_1(k_2 - k_1), c_1(k_4 - k_3), c_3(k_6 - k_5)\}$$

so that the distance h is clearly outside the kidney.

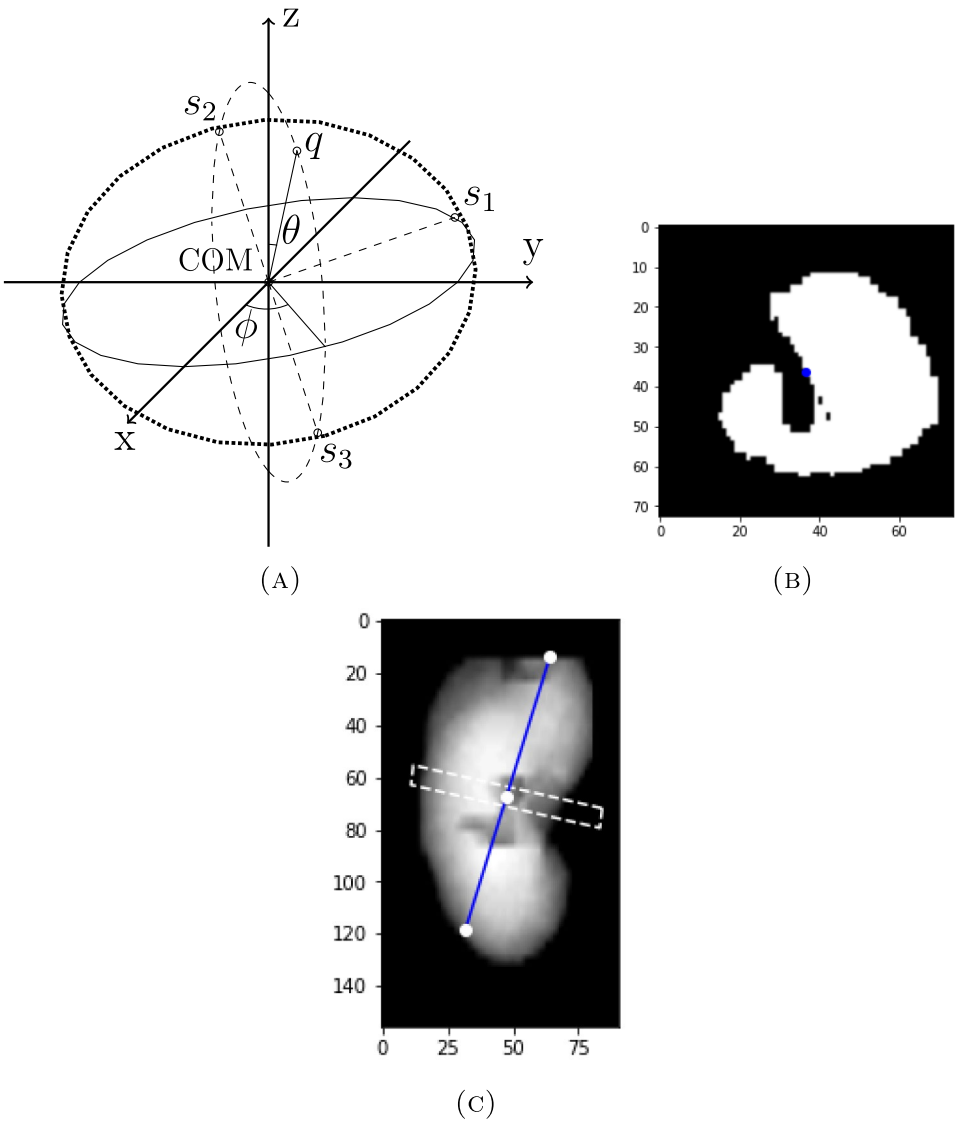


Fig. 3 **(A)** The points q, s_1, s_2, and s_3 as in (2.1) and (2.2) respect to the COM and three different circles with COM as their center and h as their radius: one on the xy-plane (solid line), one orthogonal to the *xy*-plane and passing through q (dashed line), and one orthogonal to the vector from COM to q and passing through s_1, s_2, and s_3 (densely dotted line), **(B)** the smallest cross-section area of the binary mask on a plane passing through the COM (the COM is in blue), and **(C)** the final axis (in blue) denoted on an image of the element-wise means over all the coronal slices of the binary kidney mask including COM, the axis end points, and the rectangle used to find the cross-section. The units of the axes in (**B**) and (**C**) express distance in millimeters

Next, we need to find the area of the cross-section of the binary kidney mask on the same plane as s_1, s_2, and s_3. These points can be used to define two planar vectors $u = s_2 - s_1$ and $v = s_3 - s_1$ whose lengths in millimeters rounded to the closest integer are

$$l_u = \lfloor \sqrt{(c_1 u[0])^2 + (c_2 u[1])^2 + (c_3 u[2])^2} \rceil \quad \text{and} \\ l_v = \lfloor \sqrt{(c_1 v[0])^2 + (c_2 v[1])^2 + (c_3 v[2])^2} \rceil \tag{2.4}$$

with the used voxel size $c_1 \times c_2 \times c_3$ mm^3. Here, the notation $\lfloor x \rceil$ means x rounded to the closest integer. We can find the rectangular image similar to Fig. 3(B) with $l_u \times l_v$ pixels with a pixel size of 1×1 mm^2 showing the part of the planar binary mask between the corner points $s_1, s_2, s_2 + (s_3 - s_2), s_3$ by defining the pixel at the position $[i, j]$ as the value of the point $s_1 + i/l_u \cdot u + j/l_v \cdot v$ rounded to the closest point in the original 3D binary kidney mask:

$$\begin{aligned} &\text{img}[i,j] \\ &= \text{InitialMask}[\lfloor (s_1 + iu/l_u + jv/l_v)[0] \rceil, \lfloor (s_1 + iu/l_u + jv/l_v)[1] \rceil, \\ &\lfloor (s_1 + iu/l_u + jv/l_v)[2] \rceil] \\ &\text{for} \quad i = 0, .., l_u - 1, j = 0, ..., l_v - 1. \end{aligned} \tag{2.5}$$

The area of this cross-section is calculated as the sum of this binary image, equal to the number of positive voxels in it.

Consequently, we find our axis with a loop that goes through the possible combinations of the angles $\theta \in [0, \pi/2]$ and $\phi \in [0, 2\pi)$, defines s_1, s_2, and s_3 as above, finds the related binary image, and saves its sum in a two-dimensional (2D) matrix with rows and columns related to the choices of θ and ϕ. To make the code both computationally fast and accurate, we first find one initial solution from 10×10 values for the angles and then adjust this choice for 10×10 choices near our earlier solution in two new loops. The axis with the smallest sum is our final choice and, in addition to the final choices for angles θ and ϕ, we return the the points l_0 and l_1 where the axis intersects with the boundary of the kidney VOI. An example of a resulting axis is shown in Fig. 3(C).

2.3.4 Locating the non-convex direction

After locating the kidney axis, our next task is to find the direction of the non-convex side of the kidney with respect to this axis. Namely, the area near this non-convex part contains the renal arteries and veins that we want to remove from our final segmentation. To find this non-convex, we rotate though the planes parallel to the longitudinal axis.

For any angle $\mu \in [0, 2\pi)$, the point

$$d = \text{COM} + (h, \pi/2 - \theta + 2\theta|\pi - \mu|/\pi, \phi + \mu) \tag{2.6}$$

is on the circle orthogonal to the axis defined by the angle θ and ϕ. See Fig. 4(A). To find an image showing the 2D part of the kidney mask between the kidney axis and the line parallel to axis passing through this point (2.6), we define three points

$$\begin{aligned} s_1 &= l_0 + (h, \pi/2 - \theta + 2\theta|\pi - \mu|/\pi, \phi + \mu), \\ s_2 &= l_0, \\ s_3 &= l_1 + s_1 - l_0 \end{aligned} \tag{2.7}$$

with the spherical coordinates as in (2.3). After this, we compute the image from the cross-section between these points as in (2.5), where the vectors u, v are $u = s_2 - s_1$ and $v = s_3 - s_1$ for these new points and their lengths l_u, l_v are defined as in (2.4).

For each cross-section image, we compute the area of the greatest connected component in the intersection between the complement of the cross-section of the VOI and the convex hull of this cross-section. First, we locate all the vertex points of the cross-section by searching all the positive voxels at least two of whose adjacent sides are shared with a negative voxel. Then the function *scipy.spatial.ConvexHull* is used to find the convex hull of these vertices, after which the functions *shapely.geometry.Polygon* and *rasterio.features.rasterize* are used to find a binary image of this convex hull. Finally, we create a binary image of the convex hull without the actual cross-section and the convex hull of the cross-section, find its greatest connected component, and count its area as a sum.

Similarly to the location of the axis, we find the non-convex direction with a loop that goes through the possible values of $\mu \in [0, 2\pi)$, defines s_1, s_2, and s_3, finds the binary image from these points, and saves the area of the greatest connected component in the convex hull without the cross-section in a vector. For the sake of computational efficiency, this process is actually done during two loops first of which finds an initial solution for μ and a second which adjusts it to increase the accuracy. After that, we fix the angle μ and the point d, referred below as the direction point of the kidney, based on the maximum area of this connected component. Examples of the final direction and the related connected component are shown in Fig. 4(B), (C).

2.3.5 Initial cortex mask

The next step of our algorithm is finding the initial version of the cortex mask. This is done by choosing several points inside the initial kidney VOI and finding their distance to the cortex based on the activity concentration according to the PET image. The code is built on an assumption that the maximal tracer concentration within the kidney is found at the cortex, though it removes outliers in order to return a realistic cortex masks.

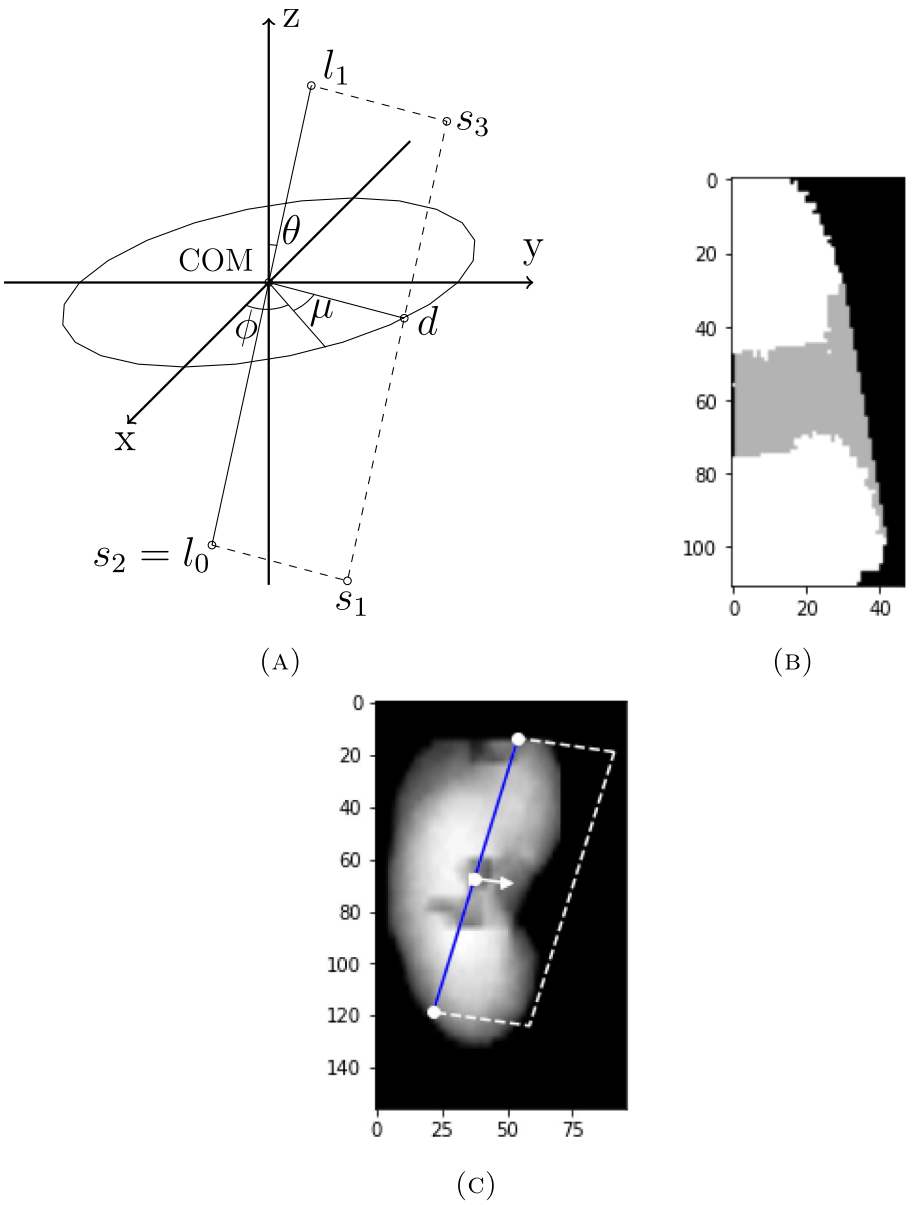


Fig. 4 **(A)** The points d, s_1, s_2, and s_3 as in (2.6) and (2.7) with respect to the axis with endpoints l_0 and l_1 and the angle μ, **(B)** the greatest connected component (in gray) outside the cross-section (in white) but inside its convex hull on a plane defined by the three points s_1, s_2, and s_3, and **(C)** the final direction denoted as an arrow on an image of the element-wise means over all the coronal slices of the binary kidney mask together with the axis (in blue) and the rectangle used to find the cross-section. The units of the axes in **(B)** and **(C)** express distance in millimeters

First, we locate two auxiliary points x and y as the vertex points with the minimum and maximum height on the cross-section image computed as in (2.5) based on the points

$$\begin{aligned} s_1 &= d + 1.3(l_0 - \mathrm{COM}), \\ s_2 &= 2\mathrm{COM} - d + 1.3(l_0 - \mathrm{COM}), \\ s_3 &= d + 1.3(l_1 - \mathrm{COM}) \end{aligned} \tag{2.8}$$

with d is the direction point defined as in (2.6) for our final choice of μ. The location of these points is shown in Fig. 5(A). After finding the points x and y in a 2D cross-section as in Fig. 5(B), we can map these back to our 3D image space by

$$[i,j] \mapsto s_1 + iu/l_u + jv/l_v$$

when $[i, j]$ is the location in the rectangular image (2.5), $u = s_2 - s_1$, $v = s_3 - s_1$, and l_u, l_v are defined as in (2.4). The resulting points x and y are on different ends of the kidney so that the piece-wise segment $[x, \mathrm{COM}, y]$ is V-shaped in an approximately similar way to the kidney, like in Fig. 5(C).

After that, we choose 100 possible center points c_i, $i = 1, ..., 100$, along the piece-wise segment $[x, \mathrm{COM}, y]$. These centers are evenly spaced and their division between the segments $[x, \mathrm{COM}]$ and $[\mathrm{COM}, y]$ is roughly according to the ratio of $|x - \mathrm{COM}|/|y - \mathrm{COM}|$. For each center c_i, we choose the radius r_i as the maximal distance from c_i to the vertex points of the cross-section along the plane perpendicular to the longitudinal kidney axis. A few examples of circles formed by these choices of c_i and r_i are shown in Fig. 5(D).

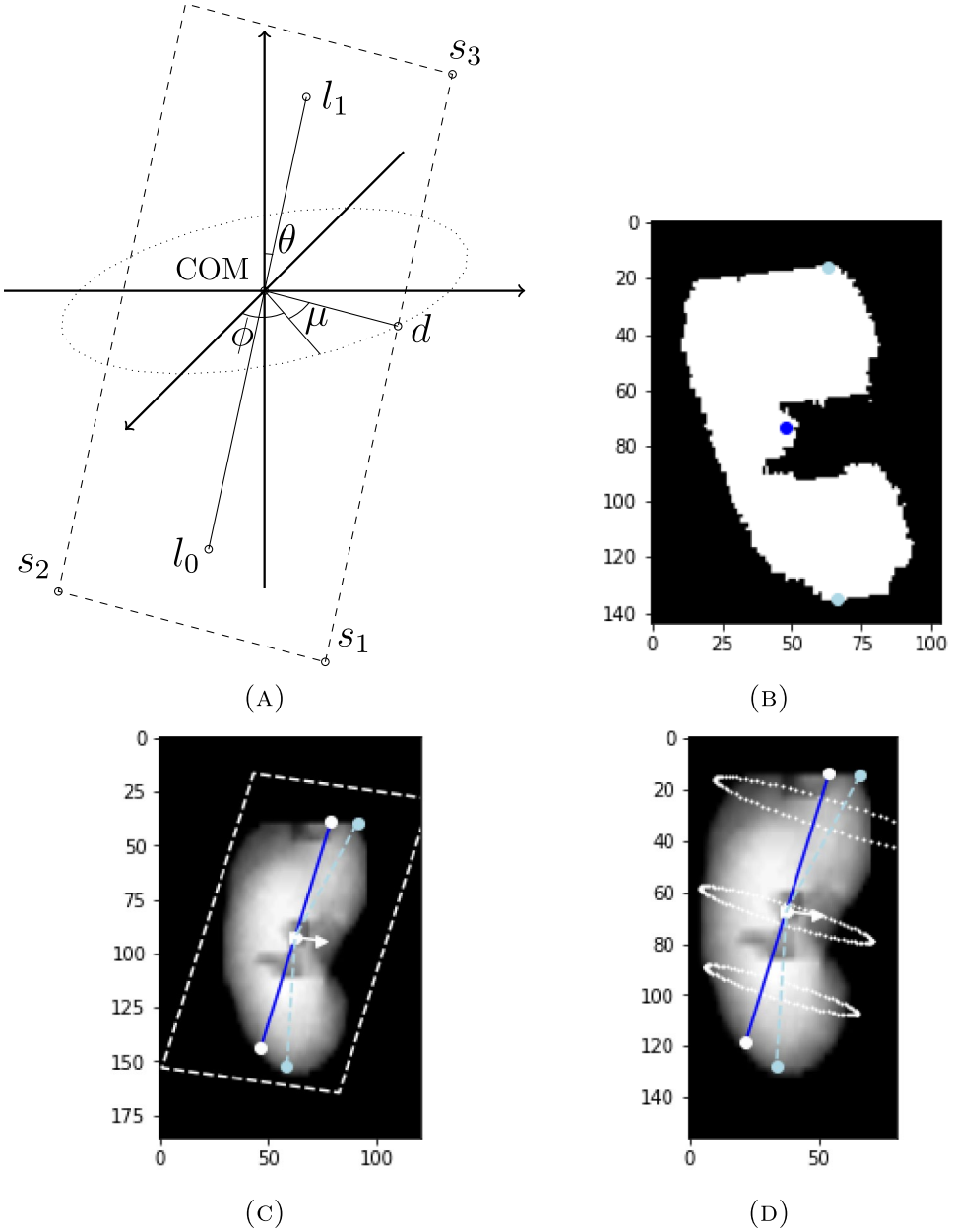


Fig. 5 **(A)** The points s_1, s_2, and s_3 as in (2.8) with respect to the axis with endpoints l_0 and l_1, the direction point d, and the angle μ, **(B)** the cross-section on a plane defined by the three points s_1, s_2, and s_3 with the points COM (blue) and x and y (in light-blue), **(C)** the line segments from x to COM and from COM to y (in dashed light-blue lines) denoted on an image of the element-wise means over all the coronal slices of the binary kidney mask together with the axis, the non-convex direction, and the rectangle used to find the cross-section, and **(D)** three circles perpendicular to the axis with their centers on $[x, \text{COM}, y]$ and radii as maximal distances to the vertex points on the cross-section parallel to the circle in question. The units of the axes in **(B)**, **(C)**, and **(D)** express distance in millimeters

Then for $j = 1, ..., 100$, we choose the radius r_i from the point c_i into the direction specified by the angle $j/100 \cdot 2\pi$ and find the values at the points spaced between 1 mm along this segment according the element-wise means of the six first time-frames of the PET image (equal to the first 25 s of the imaging period). After that, we choose the distance s_{ij} so that the sum of the values at this point s_{ij} and its two adjacent points are at maximum. We collect the information about all the distances s_{ij}, $i = 1, ..., 100$, $j = 1, ..., 100$, into a 100×100-matrix.

For each distance s_{ij}, we then compute its absolute difference d_{ij} from the distance furthest away from s_{ij} among the its four possible neighbours:

$$d_{ij} = \max\{|s_{ij} - s_{i-1,j}|, |s_{ij} - s_{i+1,j}|, |s_{ij} - s_{i,j-1}|, |s_{ij} - s_{i,j+1}|\}.$$

After this, we assess which elements s_{ij} have high enough distance d_{ij} to be considered a statistical outlier. Our criterion for outliers here is Tukey's fences, according to which a point is an outlier if its distance from the median is over 1.5 times of the interquartile range (IQR). Consequently, we create a 100×100 binary matrix that shows the locations of the distances s_{ij} whose difference d_{ij} from the nearby values is over 1.5·IQR above the median of these differences. We not only remove all these outliers from our distances s_{ij} but also go through all the connected components of the complement of our outlier set and remove them if they consist

of less than 1000 elements (10% of the 100×100-matrix). Namely, otherwise our distance matrix might include outlier distances that are surrounded by other outliers.

Finally, we create an initial version of the cortex mask by including all the points from a distance s_{ij} from a center c_i into the direction specified by the angle $j/100 \cdot 2\pi$ for the indexes i, of our remaining distances s_{ij} as positive points in a binary matrix with the same size as our choice of a bounding box. After that, we go through the slices of the initial mask in coronal, sagittal, and transaxial planes and connect the positive points of each slice to form a closed polygon as shown in Fig. 6(A), (B). This is done by first computing the mean point of all the positive points on a given slice, then computing the angle of the polar coordinate presentation of the complex vector from the mean point to a given positive point for all the positive points, and ordering the positive points based on these angles so that they can be connected to form outlines of a star-convex polygon. By connecting two points, we mean that we include all the points on the line segments between them to our initial cortex mask. Note that, even though connecting the points over all the slices in all three dimensions might cause false segmentation due to the non-convex shape of the kidney, this possible false segmentation is later removed when we fully cut out the renal vessel area.

2.3.6 Medulla and cortex separation

Our next step is to create the medulla mask and adjust the initial cortex mask. This step is relatively straight-forward: We use the information of the initial cortex mask to create a correctly placed and accurate mask for the whole kidney and then locate the medula with binary erosion.

First, we initialize a binary mask for the whole kidney VOI with the same size as our bounding box. We add the initial cortex mask as positive points. Then we determine from the end points l_0, l_1 of the longitudinal axis whether coronal, sagittal, or transaxial plane is the closest to a perpendicular plane of this axis. After that, we go through the slices of our bonding box parallel to this anatomic plane. For each slice, we choose complement of our cortex mask and add all such connected components of this complement that do not touch the sides of the bounding box to our kidney mask.

After the kidney mask has been defined, we use the binary erosion function*scipy.ndimage.morphology.binary_erosion* to divide the resulting kidney mask into two separate masks for the medulla and the cortex: The medulla mask contains all the positive voxels which survive three iterations of binary erosion meaning that their minimum distance to the boundary of the original kidney VOI includes at least three other voxels. Figure 6C shows an example of a medulla mask. Naturally, this threshold can be adjusted if needed. The final cortex mask contains those positive voxels of the initial cortex mask that are outside of the medulla.

2.3.7 Removal of renal vessel area

Our final step is the removal of the area with the renal vessels. The procedure below is repeated for both the medulla and the cortex masks.

For each positive point p of either the given mask, we find the projected point on the plane containing both the COM of the original kidney VOI used to locate the longitudinal axis and the direction point d. This projected point can be defined as

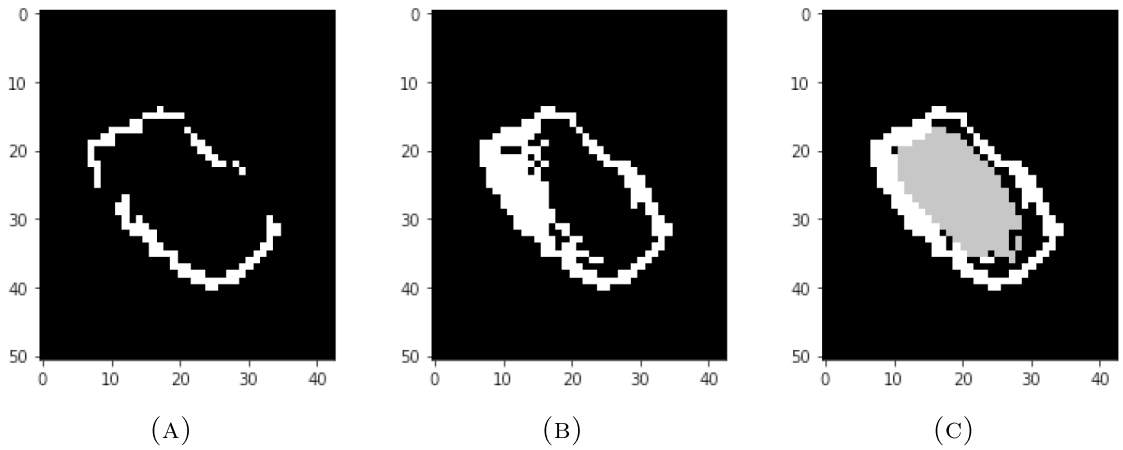


Fig. 6 A cropped transaxial slice showing the points of the initial cortex mask (**A**) before and (**B**) after the points are connected to each other, and (**C**) the cortex mask together with the medulla mask (in gray). Note that this is the 2D visualization of functions that work in 3D. The units of the axes express pixels in the transaxial slice

$$\text{projectedPoint} = \text{COM} + (p - \text{COM}) \cdot (d - \text{COM}) \frac{(d - \text{COM})}{|d - \text{COM}|^2},$$

where $\cdot$ is used for the dot product here and in (2.9). We first check if this projected point is on the same side of the axis than d or, equivalently, whether the dot product between the vectors from the COM to this projected point and from the COM to d is positive. After that, we then check if the projected point is between the points $x_1 = x + (y - x)/3$ and $y_1 = y + (x - y)/3$ when projected as

$$x_1 + (\text{projectedPoint} - x_1) \cdot (y_1 - x_1) \frac{(y_1 - x_1)}{|y_1 - x_1|^2}, \tag{2.9}$$

onto the line passing through x and y. We remove all the points that fulfill both of these conditions. Consequently, we remove the 3D space limited by three planes, one of which is parallel two axis and contains COM, one which is perpendicular to the axis and contains x_1, and one which is perpendicular to the axis and contains y_1, from our masks. The location of the points x_1 and y_1 is shown in Fig. 7(A), and the impact caused by the removal of the vessel area is visualized in Fig. 7(B), (C).

2.4 Evaluation

The algorithm was evaluated by using the data of 35 patients unseen during the design process of the algorithm.

2.4.1 Visual evaluation by expert physicians

For each 35 patient, we defined the medulla and cortex masks for both kidneys with our algorithm and plotted the outlines of the cross-sections of these masks from the middle-most coronal, sagittal, and transaxial slices of the bounding box $[(k_1 - 30) : (k_2 + 30), (k_3 - 30) : (k_4 + 30), (k_5 - 30) : (k_6 + 30)]$ on the corresponding cropped 2D slice of the PET image showing as voxel-wise mean values over time. In this way, we obtained six images from each patient showing their kidney masks from different directions. We then gave the 210 resulting images saved as.png images to an expert physician with 9 years of experience on PET imaging and manual segmentation and asked them to evaluate the quality of the segmentation. A board-certified radiologist with 25 years of experience in medical imaging confirmed these evaluations.

2.4.2 Compartment modeling

Furthermore, to assess how useful our algorithm is in compartment modeling, we inspected what kind of blood flow estimates can be obtained by using the VOIs defined by our algorithm. We computed the five different mean TACs, including one from the aorta and four other ones from the renal medulla and cortex of left and right kidney, for each patient. For computing these mean TACs, the medulla and cortex VOIs were chosen directly based on the output of our algorithm whereas the aortic VOI was the greatest connected component of the binary aorta mask given by

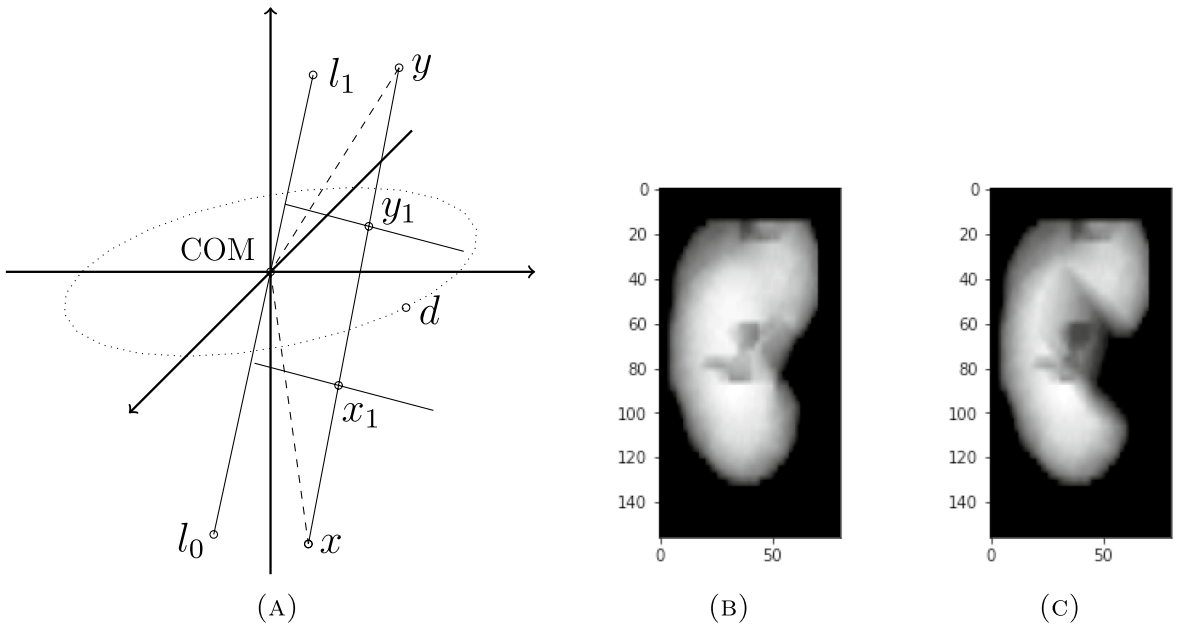


Fig. 7 **(A)** The points x_1 and y_1 with respect to x and y, the direction point d, and the axis with endpoints l_0 and l_1, and the image of the element-wise means over all the coronal slices of a binary kidney mask **(B)** before and **(C)** after removing the renal vessel area. The units of the axes in **(B)** and **(C)** express distance in millimeters

TotalSegmentator. By linear interpolation from the mean values at the original 24 time-frames, we obtained the values of the mean TACs at every second of our total scan period of 280 s.

We estimated the cortical and medullary blood flow separately for left and right kidney by fitting the standard one-tissue compartment model to each of our four tissue mean TACs. This model is defined by the differential equation

$$\frac{\partial}{\partial t}C_T(t) = k_1 C_A(t) - k_2 C_T(t), \tag{2.10}$$

where $C_T(t)$ is the tracer concentration in the medullary or cortical tissue with respect to time t, $C_A(t)$ is the image-derived input function directly estimated from the aortic mean TAC, k_1 is the rate constant of the blood flow into the tissue, and k_2 is the rate coefficient of the tracer's wash-out from the tissue. To avoid overestimation of the tissue blood flow due to the high tracer concentration in the arterial blood within the kidneys, we defined another function $C_{PET}(t)$ for the measured activity of the tissue as

$$C_{PET}(t) = (1 - V_A)C_T(t) + V_A C_A(t), \tag{2.11}$$

where V_A is the renal arterial blood volume fraction (AVF). Additionally, to account for the delay caused during the time it takes from the tracer to travel from aorta to kidney, we included a delay parameter j while model fitting.

By assuming that $C_A(i) = C_T(i) = 0$ for $i \leq 0$, we computed the values of the function $C_T(t)$ in an iterative way

$$C_T(i) = k_1 C_A(i - j - 1) + (1 - k_2)C_T(i - 1), \tag{2.12}$$

in each $n = 280$ time points of $i = 1, 2, ..., n$ seconds for any given choices of k_1, k_2, V_A, and j. We found the optimal choices of $k_1, k_2 \geq 0$ and $V_A \in (0, 1)$ for each $j = 0, 1, ..., 15$ from the initial values $k_1 = k_2 = 1$ and $V_A = 0.1$ by minimizing the model error

$$\sum_{i=1}^{n} (\tilde{C}(i) - C_{PET}(i))^2, \tag{2.13}$$

between the measured mean TAC $\tilde{C}(t)$ of the renal medulla or cortex VOI and the function $C_{PET}(t)$ defined as in (2.11) from $C_T(t)$ as in (2.12) with the non-linear Newton-type minimization algorithm *nlm* in R. As *nlm* does not allow upper or lower bounds, we forced our parameters to their desired intervals by using absolute value and the inverse-logit function as in Rainio et al. (2025). Then we fixed the value of j by choosing the one with the lowest error term (2.13).

2.4.3 Statistical testing

We used Mann–Whitney U tests to estimate whether there were statistically significant differences in medular and cortical blood flow estimates between the groups of the correct and incorrect VOIs separately for left and right kidneys. By an incorrect VOI, we mean here a VOI that was incorrect in at least one of the three slices according to the two expert physicians. Additionally, we computed the Pearson's correlation coefficient between the blood flow estimates and GFR and tested its statistical significance with a t-test. We used the standard 5% level of significance.

3 Results

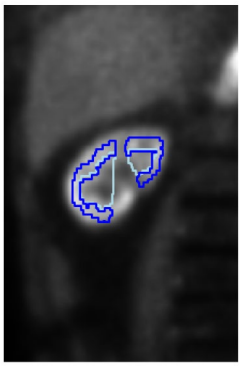
The patient characteristics of the 35 patients randomly selected for evaluation are summarized in Table 1. Additionally, two patients has impaired renal function.

Table 2 summarizes the values of the angles ϕ, θ, and μ, the length of the longitudinal kidney axis $[l_0, l_1]$, and the volumes of the medulla and cortex VOIs given by the algorithm for the 35 patients. Figure 8 shows a few of examples of these VOIs. According to the two expert physicians, the segmentation of the medulla and cortex VOIs was unsuccessful according to all the three images from the same kidney for four kidneys of four different patients out of the total of 70 kidneys of 35 patients. According the first physician, the cortex was incorrect at least some images for nine left kidneys and six right kidneys in total, typically so that the cortex was either slightly outside of the kidney or in the area of medulla. The second physician pointed out seven and five additional cases for left and right kidneys were the segmentation was not fully accurate. The kidney-level agreement on whether the segmentation by the algorithm was correct or not was 82.3% between the two physicians.

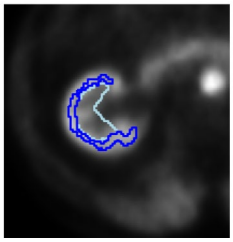
Table 3 shows the cortical and medullary blood flow of left and right kidneys estimated from the VOIs located by our algorithm. These blood flow estimates, along to the found mean TACs, are also illustrated in Fig. 9. There were no statistically significant differences caused by the inaccurate segmentation according to the Mann–Whitney U tests (p-values>0.198). The Pearson's correlation coefficient between GFR and these four blood flow types varied between 0.317−0.392, and the correlation was statistically significant between GFR and the medullary blood flow of

Table 1 The male–female sex ratio (as number of males: number of females), and mean ± standard deviation values for different variables among the 35 patients used for evaluation

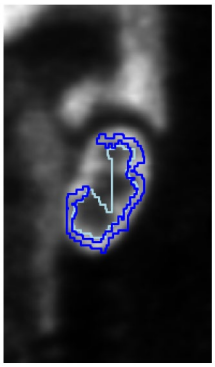
35 patients for evaluation	
M:F	19:16
Age	65.5±7.66
Height	170±8.53
Weight	87.5±19.4
BMI	30.0±5.55
GFR	85.3±18.0



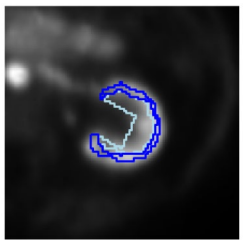
(A) Coronal, right



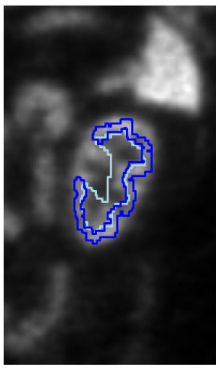
(B) Transaxial, right



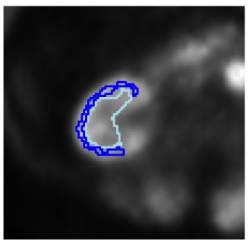
(C) Sagittal, left



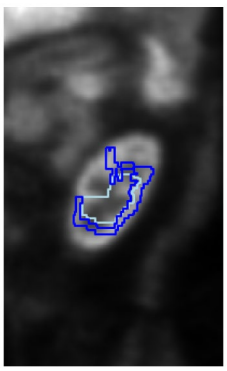
(D) Transaxial, left



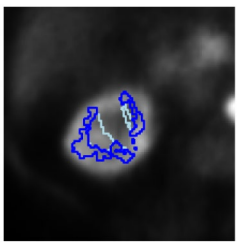
(E) Sagittal, left



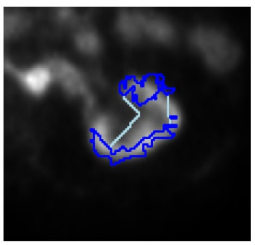
(F) Transaxial, right



(G) Sagittal, left



(H) Transaxial, right



(I) Transaxial, left

Fig. 8 Examples of 2D cross-sections of the final 3D cortex (outlined in blue) and medulla (outlined in light blue, outline often under the blue line) VOIs given by our algorithm for left and right kidneys on coronal, sagittal, or transaxial slices of one mean PET image over time. According to the doctor, the segmentation is correct for **(A)**–**(F)**, whereas the cortex VOI is in the medulla in **(G)** and **(H)** and outside kidney in **(I)**

Table 2 Mean ± standard deviation values for the angles ϕ, θ, and μ, the length of the kidney axis $[l_0, l_1]$, and volumes of medulla and cortex VOIs given by our algorithm for the left and the right kidney of the 35 patients. The angles ϕ, θ, and μ are defined as in Figs. 3(A) and 4(A) so that the positive x-axis is the anterior direction, the positive y-axis is the patient's left, and the positive z-axis is the superior direction

Kidney	ϕ (rad)	θ (rad)	μ (rad)	Axis length (cm)	Cortex VOI (cm^3)	Medulla VOI (cm^3)
Left	2.03±1.05	0.508±0.279	2.36±1.22	8.69±1.83	48.1±18.9	36.0±19.6
Right	3.98±0.786	0.519±0.251	4.55±0.698	9.09±1.82	46.4±18.5	32.7±18.3

Table 3 Mean ± standard deviation values and range between the minimum and maximum in parenthesis for cortical and medullary blood flow the left and the right kidney over the 35 patients when the 1TCM is fitted to the mean TACs defined by our algorithm

Kidney	Cortical blood flow (mL/min/mL)	Medullary blood flow (mL/min/mL)
Left	2.55±0.822 (1.22−4.95)	1.63±0.527 (0.772−3.23)
Right	2.61±0.838 (1.09−4.79)	1.63±0.484 (0.721−3.23)

left kidney and GFR (p-value=0.0400) and the medullary blood flow of right kidney (p-value=0.0324).

4 Discussion

We designed an algorithm that chooses a kidney VOI in a PET image based on initial estimate of its location, finds its longitudinal axis and the direction of the non-convex part, separates the renal cortex and medulla, and removes the renal vessel area. The purpose of this algorithm was to assist in kidney analysis, especially cortical and medullary blood flow quantification via compartment modeling. For performance evaluation of our algorithm, we asked two expert physicians to assess images of the cross-sections of cortex and medulla VOIs from different directions and

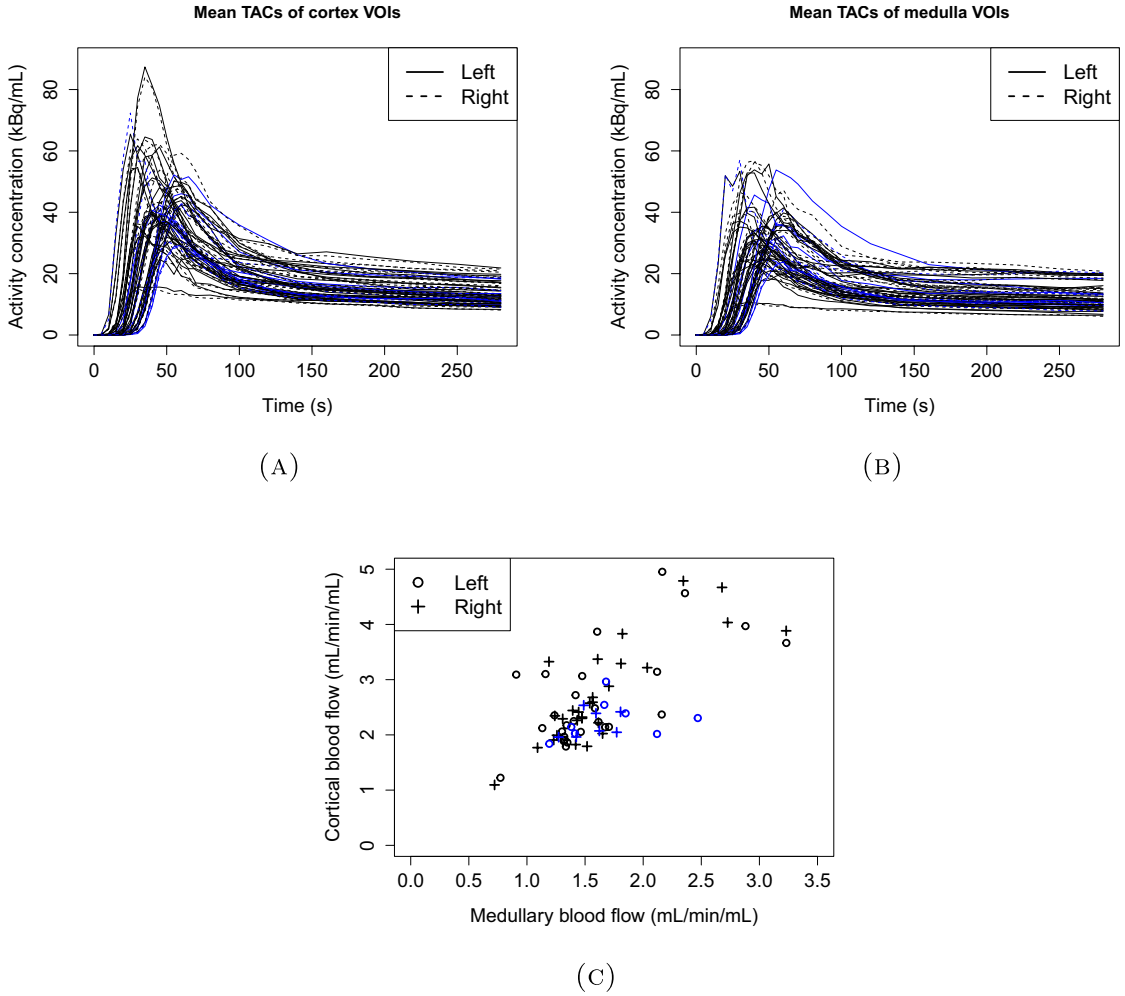


Fig. 9 Mean TACs of (**A**) cortex and (**B**) medulla VOIs (TACs from left and right kidney plotted separately) and (**C**) the estimated cortical blood flow against the medullary blood flow from the left and the right kidneys of the 35 patients. The blue color is used to indicate the cases where the VOIs were at least slightly incorrect in some directions according to both the expert physicians, whereas black is for the remaining VOIs. The one patient whose mean TACs and blood flow estimates are clearly lower than for the others patients has impaired renal function

estimated the cortical and medullary blood flow from these VOIs.

According to the evaluation by the first expert physician, the cortex and medulla VOIs were correct in all three images from different anatomic planes for 74.3% of the 35 left kidneys and 82.9% of the 35 right kidneys. Most of the segmentation errors were caused by the algorithm placing the cortex outside of the kidney or in the area of the medulla. The second physician pointed out several additional cases with potential minor segmentation errors but reported that they were such cases were it was difficult to estimate in an objective way whether the outline of the cortex had exceed the kidney or not. However, both of the physicians agreed that the segmentation by the new algorithm was correct at least in one plane of the three planes for 97.1% of the 35 left kidneys and 94.3% of the 35 right kidneys.

According our blood flow estimates, all the VOIs are accurate enough to produce the correct mean TAC for compartment modeling. Namely, it was verified by statistical testing that there were no significant differences based on the physician's assessment on the segmentation accuracy and it can be verified from Fig. 8 that there are no outliers due to our VOI selection. The likely explanation is that, unlike in typical manual segmentation where VOI is found by drawing a 2D region of interest on a few transaxial slices and combining them, our algorithm produces the VOI based on all the slices belonging to kidney. Consequently, the resulting VOI has high enough volume that its mean TAC is less sensitive to possible inaccuracies on a few slices.

Based on earlier literature summarized in Table 4, our blood flow estimates (shown in Table 3) from the found VOIs are very realistic. In particular, Rebelos et al. (2019) reported very similar blood flow estimates. Given the high age and body mass index (BMI) of our patients and their possible underlying conditions causing the symptoms that led to their referral to the imaging in the first place, our results are not directly comparable with the studies focusing only on young and lean volunteers. While the individual variation is quite high, this is also the case in the other studies.

Our algorithm could be improved further by employing some additional function that smoothens the boundaries of the cortex VOI. For instance, in Figs. 8(C) and 8(E) the cortex is very uneven, even if correctly placed. This change could be done so that the points of the initial cortex mask as in Fig. 6(A) are connected in a different method than with the line segments defining a starlike convex polygons. Alternatively, binary dilation followed by binary erosion also evens out the boundaries of a binary segment, though it might easily result in the loss of segmentation accuracy.

One limitation of our current study is that we were unable to validate our new algorithm by comparing it to another segmentation tool. To our best knowledge, there are currently no other automatic segmentation tools that separate renal medulla and cortex from PET images. However, we share all the source codes of our current method so that they could be used to develop this tool and create alternative versions.

One possible future improvement could be the application of some ML technique for post-processing. While this type of segmentation task could be fully performed with convolutional neural networks, for instance, these ML methods are often criticized for their blackbox type of decision making where the reasoning is not revealed to the user. This is also why we chose to build an algorithm in our study: Even if there were errors in the results, it is considerably easier to locate and fix them when we understand exactly how the algorithm works. Additionally, while ML methods typically require large annotated datasets, meaning that there should already be ground-truth segmentation mask of cortex and medulla VOIs available for several patients before they can be found for new patients, it was possible to create this algorithm without this data. However, in the future, we could use our algorithm to produce cortex and medulla VOIs for large number of patients, assess their quality, train a ML model with the successful masks, and use the trained ML model to perform the segmentation in the cases where our algorithm produces inaccuracies.

While our algorithm was designed with ^{15}O-water PET images, it can be used to study the perfusion of other PET

Table 4 Earlier ^{15}O-water PET studies on renal blood flow of humans in comparison with our study summarized with number *N* of subjects, whether the subjects were healthy volunteers or patients with a specific diagnosis, age in years either as mean ± standard deviation or as its range, the male–female sex ratio, the cortical and the medullary blood flow either as mean ± standard deviation or median and IQR in parenthesis, and the unit of the blood flows. RAS refers to renal artery stenosis

Research	*N*	Subjects	Age	M:F	Cortical	Medullary	Unit
Kudomi et al. (2009)	6	Healthy	58±5	NA	3.11±1.48	–	mL/min/g
Päivärinta et al. (2018)	8	Healthy	48–75	5:3	1.8±0.3	–	mL/min/g
	9	RAS bilateral	52–85	3:6	1.4±0.3		mL/min/g
Rebelos et al. (2019)	15	Healthy	45±12	0:15	2.68 (1.04)	1.63 (0.72)	mL/min/mL
	23	Morbidly obese	43±10	0:23	2.62 (0.62)	1.49 (0.51)	mL/min/mL
Our study	35	Suspected CAD	65.5±7.66	19:16	2.58±0.825	1.63±0.502	mL/min/mL

tracers in the kidney cortex and medulla. The functions of our algorithm could be also used to assist in the segmentation of organs or anatomic structures in the PET images. In particular, the function that locates the longitudinal axis of the kidney could be applied to any VOI. Additionally, the functions that produce 2D projections along a specified plane might also be of interest in the analysis of PET or other medical images. Furthermore, applying our current algorithm to different time-frames of a dynamic PET image allows us to follow its movements and avoid inaccuracies caused by respiration that is otherwise causes significant motion during the imaging (Yamashita et al. 2014).

5 Conclusion

In our study, we introduced a new kidney segmentation algorithm for cortical and medullary blood flow quantification after PET imaging. When given either an initial, possibly inaccurate kidney VOI or the location of a bounding box containing the kidney, the algorithm chooses an initial VOI, computes the longitudinal axis of kidney and the direction of its non-convex part, and finds such VOIs the renal cortex and medulla where the area of renal vessels has been removed. The performance of this algorithm was evaluated by two expert physicians visually assessing the cross-sections of cortex and medulla VOIs produced by the algorithm for left and right kidneys in the dynamic total-body ^{15}O-water PET images of 35 human patients and by applying these VOIs to estimate cortical and medullary blood flow via compartment modeling. According to the evaluation by the physicians, the cortex and medulla VOIs were at least partially correct for 94.3% of the total 70 kidneys. Additionally, the segmentation inaccuracies did not affect to the final estimates of cortical and medullary blood flow, which were very similar to the earlier research.

Funding Open Access funding provided by University of Turku (including Turku University Central Hospital). The first author was financially supported by the Otto A. Malm Foundation and Sakari Alhopuro Foundation, and the second author was supported by the Finnish Medical Foundation.

Data Availability The patient data is not available due to privacy restrictions.

Code Availability Available at github.com/rklen/Kidney_algorithm.

Declarations

Conflict of interest J.K. received consultation fees from GE Healthcare and AstraZeneca and speaker fees from GE Healthcare, Bayer, Lundbeck, Boehringer-Ingelheim, Pfizer, Merck, and Siemens, outside the submitted work.

Informed consent and ethical approval All the patients were at least 18 years of age, consented to the use of their data, and the research from their data was approved by Ethics Committee of the Hospital District of Southwest Finland.

References

Arif M et al (2020) Clinically significant prostate cancer detection and segmentation in low-risk patients using a convolutional neural network on multi-parametric MRI. Eur Radiol 30:6582–6592

Chen Gongping et al (2022) "A novel convolutional neural network for kidney ultrasound images segmentation". In: *Computer methods and programs in biomedicine* 218 , p. 106712

da Cruz Luana Batista et al (2020) "Kidney segmentation from computed tomography images using deep neural network". In: *Computers in Biology and Medicine* 123 , p. 103906

Daniel Alexander J et al (2021) "Automated renal segmentation in healthy and chronic kidney disease subjects using a convolutional neural network". In: *Magnetic resonance in medicine* 86.2 , pp. 1125–1136

Elsayed Essam F et al (2007) "Cardiovascular disease and subsequent kidney disease". In: *Archives of internal medicine* 167.11 , pp. 1130–1136

Florkowski Christopher M, Chew-Harris Janice SC (2011) "Methods of estimating GFR–different equations including CKD-EPI". In: *The Clinical Biochemist Reviews* 32.2 , p. 75

Gillies Sean et al (2007) "Shapely: Manipulation and analysis of geometric objects in the Cartesian plane". In: URL: https://pypi. org/project/Shapely/[accessed 2021-04-08]

Gillies Sean et al (2013) *Rasterio: geospatial raster I/O for Python programmers*. Mapbox, URL: https://github.com/rasterio/rasterio

Guo Junyu, Odu Ayobami, Pedrosa Ivan (2022) "Deep learning kidney segmentation with very limited training data using a cascaded convolution neural network". In: *PloS one* 17.5 , e0267753

Haghighi Marzieh, Warfield Simon K, Kurugol Sila (2018) "Automatic renal segmentation in DCE-MRI using convolutional neural networks". In: 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018). IEEE. 1534–1537

Knuuti Juhani et al (2023) "Quantitative perfusion imaging with total-body PET". In: *Journal of Nuclear Medicine* 64.Supplement 2 , 11S–19S

Kudomi N et al (2009) Parametric renal blood flow imaging using [15 O] H 2 O and PET. Eur J Nucl Med Mol Imaging 36:683–691

Kurata Yasuhisa et al (2019) "Automatic segmentation of the uterus on MRI using a convolutional neural network". In: *Computers in biology and medicine* 114 , p. 103438

Liu Z et al (2020) Segmentation of organs-at-risk in cervical cancer CT images with a convolutional neural network. Physica Med 69:184–191

Osadebey M et al (2021) Three-stage segmentation of lung region from CT images using deep neural networks. BMC Med Imaging 21:1–19

Päivärinta J et al (2018) The renal blood flow reserve in healthy humans and patients with atherosclerotic renovascular disease measured by positron emission tomography using [15 O] H 2 O. EJNMMI Res 8:1–7

Park B et al (2019) Lung segmentation on HRCT and volumetric CT for diffuse interstitial lung disease using deep convolutional neural networks. J Digit Imaging 32:1019–1026

Pereira Sérgio et al (2016) "Brain tumor segmentation using convolutional neural networks in MRI images". In: *IEEE transactions on medical imaging* 35.5 , pp. 1240–1251

R Core Team et al (2013) *R: A language and environment for statistical computing*

Rainio Oona et al (2023) "Carimas: an extensive medical imaging data processing tool for research". In: *Journal of Digital Imaging* 36.4 , pp. 1885–1893

Rainio Oona, Klén Riku (2025) "Compartmental modeling for blood flow quantification from dynamic 15 O-water PET images of humans: a systematic review". In: *Annals of Nuclear Medicine* , pp. 1–16

Rainio Oona, Knuuti Juhani, Klén Riku (2025) "Investigation of total-body applicable compartment models for kinetic modeling in total body 15 O-water perfusion PET imaging". In: *Network Modeling Analysis in Health Informatics and Bioinformatics* 14.1 , p. 22

Rebelos Eleni et al (2019) "Renal hemodynamics and fatty acid uptake: effects of obesity and weight loss". In: *American Journal of Physiology-Endocrinology and Metabolism* 317.5 , E871–E878

Rossum Guido Van, Drake Fred L (2009) *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, . ISBN: 1441412697

Thomas Merlin C et al (2015) "Diabetic kidney disease". In: *Nature reviews Disease primers* 1.1 , pp. 1–20

Townsend DW et al (2004) "Physical principles and technology of clinical PET imaging". In: *Annals-Academy of Medicine Singapore* 33.2 , pp. 133–145

Virtanen Pauli et al (2020) "SciPy 1.0: fundamental algorithms for scientific computing in Python". In: *Nature methods* 17.3 , pp. 261–272

Wasserthal Jakob et al (2023) "TotalSegmentator: robust segmentation of 104 anatomic structures in CT images". In: *Radiology: Artificial Intelligence* 5.5

Wright-Nunes Julie A et al (2012) "Patient knowledge of blood pressure target is associated with improved blood pressure control in chronic kidney disease". In: *Patient education and counseling* 88.2 , pp. 184–188

Yamashita H et al (2014) Individually wide range of renal motion evaluated by four-dimensional computed tomography. Springerplus 3:1–7

Yu Qian et al (2019) "Crossbar-net: A novel convolutional neural network for kidney tumor segmentation in ct images". In: *IEEE transactions on image processing* 28.8 , pp. 4060–4074