

# Реконструкция трёхмерной геометрии коронарных артерий\*

**В. В. БОРИСЕНКО**

*Московский государственный университет  
им. М. В. Ломоносова  
e-mail: vladimir\_borisen@mail.ru*

**Т. Н. ВЕСЕЛОВА**

*ФГБУ «НМИЦ кардиологии» Минздрава России  
e-mail: tnikveselova@gmail.com*

**С. К. ТЕРНОВОЙ**

*ФГАОУ ВО Первый МГМУ им. И. М. Сеченова Минздрава России  
(Сеченовский Университет),  
ФГБУ «НМИЦ кардиологии» Минздрава России  
e-mail: prof\_ternovoy@list.ru*

**А. М. ЧЕПОВСКИЙ**

*Национальный исследовательский университет  
«Высшая школа экономики»  
e-mail: achepovskiy@hse.ru*

УДК 004.932:004.925

**Ключевые слова:** компьютерная томография, триангуляция поверхности, маршрутирующие тетраэдры, алгоритм роста области из семени, сглаживание поверхности методом Subdivision Surface.

## Аннотация

Рассматривается задача построения триангуляции внутренней поверхности аорты и коронарных артерий. Это необходимо для создания гидродинамической модели кровоснабжения сердца. Исходными данными являются поперечные томографические срезы сердца конкретного пациента толщиной 0,5 мм. Триангуляцию аорты и коронарных артерий планируется использовать для расчёта кровотока в программном комплексе FlowVision, предназначенном для моделирования трёхмерных течений жидкости и газа. Результаты моделирования планируется применять в практической медицине для диагностики ишемии миокарда.

Триангуляция вычисляется с помощью трёхмерного алгоритма роста области из семени, который создаёт множество вокселей. Затем строится триангуляция поверхности этого множества, и на последнем этапе применяется алгоритм сглаживания Subdivision Surface. Также описывается полуавтоматический метод определения области интереса, что необходимо для отделения аорты и коронарных артерий от внутренней области сердца, вен и других структур, которые не должны быть включены в построенную триангуляцию.

---

\*Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 18-29-26012.

### Abstract

*V. V. Borisenko, T. N. Veselova, S. K. Ternovoy, A. M. Chepovskiy, Reconstruction of 3D geometry of coronary arteries, Fundamentalnaya i prikladnaya matematika, vol. 23 (2020), no. 2, pp. 37–56.*

We consider the problem of 3D reconstruction for the internal surface of aorta and coronary arteries. This is needed for the creation of a hydrodynamic model of blood supply to the heart. The initial data are transverse tomographic sections of the heart of a particular patient with a thickness of 0.5 mm. Triangulation of the aorta and coronary arteries is to be used to calculate a blood flow in the software package FlowVision, designed for modeling three-dimensional flows of liquid and gas. The simulation results are planned to be applied in practical medicine for the diagnosis of myocardial ischemia.

Triangulation is calculated using a three-dimensional Seeded Region Growing algorithm that creates a set of voxels. Then the triangulation of the surface of this set is constructed and smoothed by the Subdivision Surface method. We also describe a semi-automatic procedure to define a Region of Interest used to separate the aorta and coronary arteries from the inner volume of heart, veins and other structures that should not be included in the constructed triangulation.

## Введение

Ранняя диагностика заболеваний, связанных с ухудшением кровоснабжения сердца, играет важную роль в профилактике острого инфаркта миокарда и внезапной коронарной смерти. Снижение коронарного кровотока чаще всего обусловлено прогрессирующим ростом на сосудистой стенке атеросклеротической бляшки, которая вызывает сужение (стеноз) просвета артерий, снабжающих кровью сердечную мышцу. Для оценки состояния коронарного русла применяют инвазивное внутрисосудистое исследование — коронарную ангиографию (КАГ) и неинвазивную компьютерную томографическую ангиографию (КТА). Для получения изображений коронарных артерий с помощью этих методов используется рентгеновская трубка. Визуализация сердца и сосудов возможна только при введении в кровь пациента рентгеноконтрастного средства: при КАГ катетер проводится непосредственно к артериям сердца, при КТА устанавливается в периферической вене. Традиционная КАГ позволяет визуализировать внутренний просвет сосуда, при КТА возможно оценить не только просвет артерии, но и состояние сосудистой стенки и миокарда [4, 5, 9]. Современные компьютерные томографы позволяют синхронизировать процесс получения поперечных томографических срезов сердца с сердечным ритмом, сбор данных занимает всего 0,35 с, что значительно минимизирует артефакты от сердечных сокращений. Программное обеспечение компьютерного томографа позволяет реконструировать томографические срезы в разных сечениях без потери качества изображений. В частности, можно воссоздать серию сечений, перпендикулярных к оси сосуда, и в каждом сечении определить поперечный контур и площадь сосуда, толщину сосудистой стенки. Это даёт возможность вычислить площади поперечных сечений в серии равномерно распределённых точек вдоль оси

сосуда и рассчитать объём проходящей крови, перепад давлений и другие характеристики. Однако расчёты при таком методе достаточно приближительные, не учитывается, например, конкретная форма сосуда.

Основная идея описываемого проекта состоит в том, чтобы полностью и точно воссоздать модель кровотока методами гидродинамики, применяя численные методы решения дифференциальных уравнений, описывающих течение жидкости (уравнения Навье—Стокса, уравнение состояния и др.). В области аэро- и гидродинамики накоплен громадный опыт применения численных методов для расчёта автомобилей, самолётов, кораблей, газовых турбин и т. п. Разработанные методы можно применить и в медицине для изучения кровообращения, причём модель кровообращения строится и рассчитывается для каждого конкретного пациента с учётом особенностей его кровеносной системы, для чего используются результаты КТА аорты и коронарных артерий. Мы изучаем кровоснабжение сердца и строим модель кровотока в аорте и коронарных артериях.

Для гидродинамического расчёта предполагается использовать программный комплекс FlowVision [1, 2], представляющий собой развитую программную среду для моделирования трёхмерных течений жидкости и газа в самых разных областях применения. Для решения уравнений динамики жидкости и газа FlowVision строит расчётную сетку с локальным измельчением расчётных ячеек. При этом трёхмерная геометрия расчётной области задаётся путём импорта модели поверхности из систем САПР или конечно-элементных систем. Такая модель представляет собой триангуляцию поверхности расчётной области и импортируется в виде VRML- или STL-файла.

В данной работе рассматривается построение трёхмерной триангуляции внутренней поверхности аорты и коронарных артерий сердца. Эта триангуляция предназначена для импорта в программный комплекс FlowVision и последующего расчёта кровотока в области сердца. Исходными данными для построения триангуляции являются результаты томографического обследования конкретного пациента.

## Компьютерная томография

Преобразование Радона, которое лежит в основе компьютерной томографии, было известно ещё в начале XX века. Оно позволяет заглянуть внутрь непрозрачного предмета, т. е. воссоздать сечение предмета по интегралам вдоль различных прямых линий в этом сечении. Практическое применение преобразования Радона стало возможным только после возникновения компьютеров и их усовершенствования. Компьютерный томограф использует источник рентгеновского излучения — рентгеновскую трубку, которая вращается в фиксированной плоскости вокруг тела пациента. На противоположной стороне вращающегося кольца расположен массив приёмников рентгеновского излучения, который вращается синхронно с источником излучения. Каждый из приёмников принимает луч, прошедший через сканируемый объект вдоль прямой, соединяющей

рентгеновскую трубку и приёмник, т. е. определяет интеграл от плотности объекта вдоль этой прямой. Эти интегралы передаются томографом работающему в связке с ним компьютеру, который выполняет преобразование Радона для воссоздания плотности объекта в каждой точке сечения.

Термин «плотность объекта» не совсем точный: это не традиционная плотность в  $\text{г/см}^3$ , а степень поглощения рентгеновских лучей веществом в данной точке сканируемого объекта. Единицы измерения называются единицами Хаунсфилда, сокращённо HU (Hounsfield Unit). Величина HU является целым числом со знаком. Для вакуума (или воздуха)  $\text{HU} = -1024$ , для воды  $\text{HU} = 0$ , наиболее плотные кости и металлические включения имеют HU от 500 до нескольких тысяч. Для человеческого тела ткань с наименьшей плотностью — это лёгкие, в среднем  $\text{HU} \approx -850$ , для жировой ткани  $\text{HU} \approx -115$ , мышечная ткань имеет значения HU от +35 до +55, плотность костей варьируется в очень широких пределах, от 150 для пористых и тонких костей до  $\text{HU} > 1000$  для твёрдых костей и зубов.

Отметим, что томограф выдаёт не изображения (как многие ошибочно считают), а матрицы плотности исследуемого объекта в серии сечений, чаще всего перпендикулярных вертикальной оси тела пациента (такие сечения называются аксиальными); эти матрицы обычно называют томографическими срезами. Для современных томографов типичный размер матриц равен  $512 \times 512$ , хотя в ближайшем будущем, возможно, томографы уже будут выдавать матрицы размера  $1024 \times 1024$ .

Чтобы превратить матрицу плотностей в изображение, надо задать отображение единиц плотности HU на шкалу серых цветов. Традиционно это делается с помощью задания двух параметров: WC (Window Center) и WW (Window Width). Центр окна WC задаёт значение HU, которое отображается на серый цвет с интенсивностью 50 %. Значения  $\text{HU} = \text{WC} + \text{WW}/2$  и более высокие отображаются на белый цвет, значения  $\text{HU} = \text{WC} - \text{WW}/2$  и более низкие — на чёрный. В пределах HU от  $\text{WC} - \text{WW}/2$  до  $\text{WC} + \text{WW}/2$  (размер этого интервала равен ширине окна WW) значения плотности линейно отображаются на шкалу серых цветов от чёрного до белого.

Задавая разные значения WC/WW, мы можем изучать различные ткани на одном и том же томографическом срезе. Например, если нас интересует лёгочная ткань, то мы задаём параметры  $\text{WC} = -400$ ,  $\text{WW} = 1600$  (интервал серых цветов от  $-1200$  до  $400$ ), при этом другие ткани с  $\text{HU} \geq 400$  (костная ткань, сосуды и внутренний объём сердца при использовании контрастного средства) изображаются белым цветом независимо от величины HU. Если мы на том же томографическом срезе хотим рассмотреть мягкие ткани и кровеносные сосуды (при использовании контрастного средства), то мы задаём  $\text{WC} = 250$ ,  $\text{WW} = 800$  (интервал серых цветов от  $-150$  до  $650$ ). При этом уже лёгочная ткань со значениями  $\text{HU} < -150$  будет отображаться на чёрный цвет, т. е. вообще не будет видна (рис. 1).

Для каждого типа ткани рентгенолог подбирает наиболее подходящие параметры WC/WW, причём их можно плавно менять (обычно движением мыши

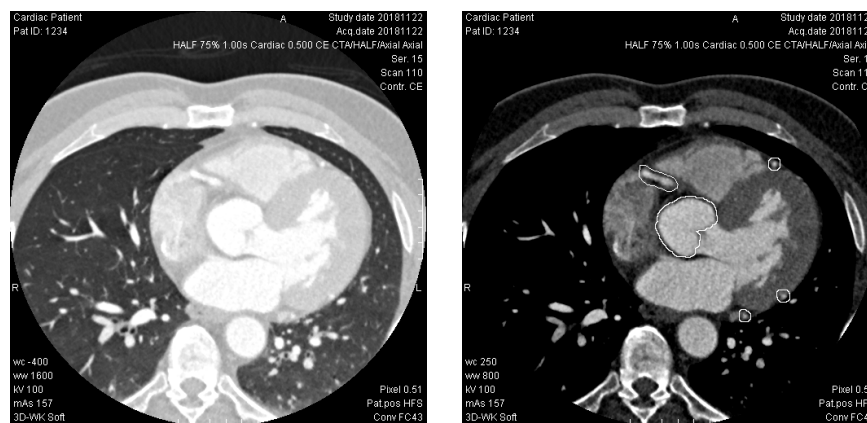


Рис. 1. Компьютерная томография органов грудной клетки с внутривенным контрастированием. Изображения одного и того же пациента в «лёгочном окне» (слева) и «мягкотканном окне» (справа). Для «лёгочного окна» используются параметры  $WC = -400$ ,  $WW = 1600$ , для «мягкотканного окна» —  $WC = 250$ ,  $WW = 800$ . На изображении в «мягкотканном режиме» контурами выделены аорта и коронарные артерии

с нажатой левой клавишей: перемещение мыши в вертикальном направлении меняет  $WC$ , в горизонтальном —  $WW$ , при этом плавно меняется и изображение томографического среза).

Отметим, что оценить просвет кровеносных сосудов на бесконтрастных изображениях невозможно, так как рентгеновская плотность неконтрастированной крови идентична плотности бляшки или тромба, которые вызывают сужение или закупорку сосуда. Поэтому для оценки состояния сосудов в кровь вводят йодсодержащее рентгеноконтрастное средство, т. е. вещество, значительно повышающее степень поглощения кровью рентгеновских лучей и безвредное для человека. В результате рентгеновская плотность крови повышается в среднем до значений от 200 до 500 и кровеносные сосуды становятся максимально «яркими» на фоне окружающих мягких тканей (лёгких, мышц, жировой клетчатки).

## Методы построения трёхмерных моделей на основе результатов компьютерной томографии

Чаще всего компьютерный томограф воссоздаёт серию параллельных сечений тела пациента. Сечения эти обычно перпендикулярны вертикальной оси тела пациента, хотя возможны и сечения в наклонной плоскости. Расстояния между сечениями могут варьироваться, но чаще всего они задаются постоянными (в нашем случае это шаг 0,5 мм вдоль оси  $Z$ ). Для решения сформулированной задачи предполагается, что сечения перпендикулярны оси тела пациента и

расстояния между соседними сечениями одинаковы. Полученные в результате сканирования матрицы задают функцию плотности тела  $d(x, y, z)$  в трёхмерном пространстве; для вычисления этой функции в точках между сечениями используется интерполяция.

Отметим, что в стандарте создания, хранения, передачи и визуализации цифровых медицинских изображений DICOM оси координат всегда связаны с телом пациента: ось  $X$  направлена от правой руки пациента к левой, ось  $Y$  — от груди к спине, ось  $Z$  — от ног к голове; единицы измерения длины — всегда миллиметры. Файлы или данные, передаваемые внутри сети по протоколам DICOM, обязательно содержат, помимо матриц пикселей, ещё много другой информации: размеры матриц, геометрические размеры пикселя в миллиметрах, позицию и ориентацию сечения в указанной системе координат, время и дату получения среза, имя и идентификатор пациента, параметры исследования и серии срезов и т. д. Файл или набор данных DICOM состоит из последовательности элементов. Номер элемента задаётся двумя беззнаковыми 16-разрядными числами, которые содержатся в словаре DICOM и обычно записываются в шестнадцатеричной системе счисления; каждый элемент содержит номер, тип данных, длину поля данных и собственно данные. Например, элемент «Имя пациента» содержит номер (0010,0010), тип «PN» (Personal Name), длину имени в байтах и собственно имя. Поддержка стандарта DICOM в настоящее время обязательна для любого медицинского оборудования и программного обеспечения.

Задача трёхмерного восстановления формулируется как построение триангуляции поверхности уровня для функции плотности. Допустим, требуется построить триангуляцию поверхности кости. Костная ткань имеет HU от 150 и выше. Примем величину 150 в качестве порогового значения. Тогда искомая поверхность задаётся уравнением

$$f(x, y, z) = d(x, y, z) - 150 = 0,$$

где  $d(x, y, z)$  — функция плотности.

Имеются два основных подхода к решению этой задачи: ячеечные методы и трёхмерные алгоритмы роста области.

## Ячеечные методы построения триангуляции

В ячейчных методах пространство разбивается на ячейки небольшого размера, представляющие собой многогранники, заполняющие без пропусков трёхмерное пространство. Значения функции  $f(x, y, z)$  вычисляются только в вершинах многогранников. Предполагается, что на рёбрах функция  $f$  меняется линейно. Если в двух вершинах ребра функция имеет значения разных знаков, то вычисляется точка на ребре, в которой функция равна нулю. Эта точка добавляется к множеству вершин триангуляции. Внутри многогранника полученные точки на рёбрах соединяются одним или несколькими треугольниками (рис. 2).

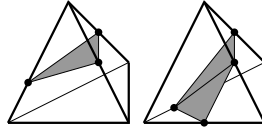


Рис. 2. Построение триангуляции внутри тетраэдрической ячейки. При трёх точках на рёбрах они соединяются одним треугольником, при четырёх — двумя, других случаев не бывает

Исторически самым первым подобным методом был алгоритм *Marching Cubes* (англ. «марширующие кубы») [18] (1987), в котором пространство разбивается на кубические ячейки. У этого метода, однако, есть серьёзные недостатки. Всего с точностью до симметрии имеется 14 различных случаев расположения вершин триангуляции на рёбрах куба (если не учитывать симметрию, то случаев  $2^8 - 2 = 254$ ). При этом в пяти случаях можно соединить точки треугольниками различными способами. При несогласованном выборе триангуляции внутри соседних кубов в построенной поверхности могут возникнуть дыры.

Для решения этой проблемы Е. Черняевым [12] (1995) была предложена коррекция этого метода, в которой выбор между несколькими вариантами триангуляции внутри куба производится на основе вычисления квадратичной формы на стороне куба; при этом и без того непростой алгоритм усложняется ещё сильнее (вместо 256 случаев в исходном алгоритме приходится рассматривать 760).

Сложности алгоритма *Marching Cubes* исчезают при использовании *тетраэдрических* ячеек вместо кубических. Топологически имеется лишь два разных случая расположения вершин триангуляции на рёбрах тетраэдра (без учёта симметрии — 14 случаев вместо 254 для куба) (см. рис. 2). При этом вообще нет проблемы несогласованного выбора вариантов триангуляции внутри смежных тетраэдров. Единственная сложность состоит в построении тетраэдрической сетки. В отличие от плоскости, которую можно разбить на правильные треугольники, трёхмерное пространство невозможно разбить на правильные тетраэдры. Поэтому возникает проблема выбора наилучшей тетраэдрической сети.

В большинстве графических пакетов, использующих тетраэдрические сети, пространство сначала разбивается на кубики и затем каждый кубик — на тетраэдры. Без добавления дополнительных вершин куб можно разбить на пять или шесть тетраэдров. При разбиении на пять тетраэдров (метод Канейро [10]) в середину куба вписывается правильный тетраэдр, рёбра которого являются диагоналями шести граней куба, объём этого тетраэдра равен  $1/3$  от объёма куба. Грани вписанного тетраэдра отсекают от куба ещё четыре тетраэдра, каждый объёмом  $1/6$ . Разбиения на тетраэдры соседних кубов в методе Канейро являются зеркально-симметричными.

На шесть тетраэдров (каждый объёмом  $1/6$ ) куб можно разбить пятью различными способами с точностью до симметрии. В [13] был рассмотрен самый

удобный из таких способов, он носит название МТ6 (Marching Tetrahedra 6). Преимущество этого способа перед методом Канейро состоит в том, что, во-первых, все шесть тетраэдров равны между собой, во-вторых, построенная сетка инвариантна относительно сдвигов на величину ребра куба в трёх ортогональных направлениях (нет необходимости чередовать зеркально-симметричные разбиения внутри смежных кубов). Недостаток метода в том, что качество тетраэдров в этом методе чуть хуже, чем в методе Канейро. (Мерой качества тетраэдра является параметр «аспект», равный отношению радиусов описанной и вписанной сфер. Аспект правильного тетраэдра равен 3, для любого другого тетраэдра он больше трёх. Чем больше аспект, тем сильнее тетраэдр отличается от правильного и тем ниже его качество.)

Разбиения куба на пять и шесть тетраэдров в алгоритмах Канейро и МТ6 показаны на рис. 3. В данной работе программно реализован именно метод МТ6.

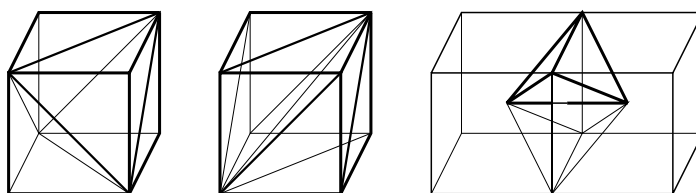


Рис. 3. Слева и в середине показаны разбиения куба на пять и шесть тетраэдров в методах Канейро и МТ6. Справа показан процесс построения тетраэдров в методе Скалы

Наилучшее качество тетраэдров достигается в методе Скалы [20] (2000), который уже не сводится к разбиению кубов на тетраэдры. Вершины тетраэдров в нём расположены в вершинах кубической решётки, а также в центрах тех же кубов. Отрезок, соединяющий центры каждых двух смежных кубов, образует четыре тетраэдра со скрещивающимися рёбрами, общими для этих двух кубов (см. рис. 3). Построенные таким образом тетраэдры заполняют без промежутков всё пространство. Вершины тетраэдров в методе Скалы соответствуют расположению атомов углерода в кристалле алмаза. Смежные тетраэдры зеркально-симметричны относительно их общей грани. Такие разбиения пространства на зеркально-симметричные тетраэдры (их теперь называют тетраэдрами Кокстера) были описаны советским математиком Р. Н. Бончковским [3] ещё в 1935 г. Он показал, что всего существует четыре таких разбиения, среди них есть и разбиение Скалы. Впоследствии этот результат был обобщён Кокстером [8] на случай пространства произвольной размерности и любых многогранников, образующих симметричную сетку, инвариантную относительно отражений (иногда такие структуры называют калейдоскопами [6]).



## Трёхмерный алгоритм роста области из семени

Традиционно алгоритм роста области чаще всего используется в двумерном случае для сегментации изображений; этот алгоритм носит название Seeded Region Growing (англ. «рост области из семени»). Алгоритм работает с множеством  $S$  пикселей изображения, которое растёт в процессе выполнения алгоритма. Вначале это множество состоит из одного или нескольких пикселей («семян»), выбранных каким-либо способом внутри интересующей нас области. Для работы алгоритм использует какую-либо динамическую структуру, также содержащую пиксели: очередь, стек и т. п. (Наиболее удобной структурой является дек, двусторонняя очередь, входящая в стандартную библиотеку языка C++.) Обозначим эту вспомогательную структуру через  $D$ . Рассмотрим вариант с единственной начальной точкой  $p_0$ . В алгоритме при добавлении точки к множеству  $S$  мы одновременно помещаем её и в очередь  $D$ . Затем, если очередь не пуста, мы извлекаем точку из начала очереди и рассматриваем всех её соседей  $n$ . Если сосед  $n$  уже принадлежит множеству  $S$ , то мы его пропускаем. В противном случае проверяется, удовлетворяет ли  $n$  критерию, определяющему область (например, интенсивность пикселя больше порогового значения). Если да, то  $n$  добавляется в  $S$  и одновременно помещается в конец очереди  $D$ . Алгоритм заканчивается, когда очередь становится пустой. Вот запись этого алгоритма на псевдокоде.

```

S: множество пикселей; D: очередь пикселей (дек)
p0: начальный пиксель («семя»)
S.добавить(p0)
D.добавить в конец (p0)
цикл пока D не пуста выполнять
|   t = D.начало; D.удалить начало
|   цикл для каждого соседа n пикселя t выполнить
|   |   если n не принадлежит множеству S и n удовлетворяет
|   |   |   критерию принадлежности к области
|   |   |   то S.добавить(n)
|   |   |   D.добавить в конец(n)
|   |   конец если
|   конец цикла
конец цикла

```

В качестве соседей пикселя берутся либо четыре прилегающих пикселя (только одна из координат отличается на единицу), либо восемь пикселей (любая из координат может отличаться на единицу).

Трёхмерный вариант алгоритма роста области практически ничем не отличается от двумерного (см. [14]). Вместо пикселей используются воксели (volumetric pixels), которые можно представлять себе как маленькие кубики

в пространстве. Положение вокселя задаётся тремя целочисленными координатами  $(x, y, z)$ , причём в случае данных томографии в качестве  $z$  используется номер среза. Разные варианты алгоритма рассматривают либо 6, либо 26 соседних вокселей. Множество вокселей эффективно реализуется как массив битовых матриц, на каждый воксель расходуется всего один бит памяти. Поскольку при компьютерной томографии элементами матриц срезов являются знаковые 16-разрядные числа, для хранения множества вокселей требуется памяти в 16 раз меньше, чем для исходных данных.

В результате работы алгоритма мы получаем множество вокселей, составляющих искомую область. Нам, однако, необходимо построить триангуляцию её поверхности. Это нетрудно сделать, исходя из представления вокселей как кубиков в пространстве. Поверхность области состоит из внешних граней кубиков, к которым не прилегают другие кубики. Каждая внешняя грань делится диагональю на два треугольника, и мы получаем искомую триангуляцию.

### Операторы математической морфологии

Алгоритмы роста области, как двумерный, так и трёхмерный, имеют следующий недостаток. Всегда матрицы исходных данных (изображения или поперечные срезы) содержат некоторый шум, т. е. значения пикселей могут иметь случайные отклонения. Обычно в любых алгоритмах трёхмерного восстановления применяется сглаживающий (гауссовский) фильтр, но этого всё равно недостаточно. Построенное множество вокселей может иметь отверстия в середине, негладкую границу и т. п. Для устранения подобных дефектов обычно применяют операторы математической морфологии (см. [19]). Есть два базовых оператора: Dilation (раздутие области) и Erosion (сужение). Используются

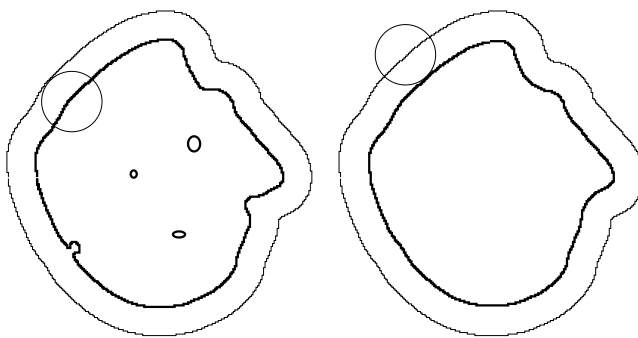


Рис. 4. Применение оператора математической морфологии Closing к двумерной области. Слева к исходной области применяется оператор Dilation, «раздувающий» её с помощью круглого структурного элемента. Справа к раздутой области применяется оператор Erosion, сужающий область. В результате граница области сглаживается и отверстия внутри области закрываются

также два «составных» оператора. Оператор Closing (закрытие области) состоит в последовательном применении сначала оператора Dilation и затем Erosion; оператор Opening, наоборот, сначала применяет Erosion и затем Dilation. Область раздувается или сужается с помощью *структурирующего элемента*, обычно это круг или шар в трёхмерном случае. Чем больше его радиус, тем больше раздувается или сужается область при выполнении операторов Dilation и Erosion.

Применение составного оператора Closing приводит к закрытию отверстий и пустот в области, а также к сглаживанию её границы (рис. 4). Чем больше радиус структурирующего элемента, тем больше радиус отверстий, которые закрываются в результате его применения. Оператор Opening, наоборот, не закрывает отверстия, а отрезает «щупальца», растущие из области. Для наших целей оператор Opening не используется.

### **Сравнение двух типов алгоритмов трёхмерного восстановления**

Алгоритмы ячеечного типа более универсальные, с их помощью можно реконструировать поверхности как сплошных объектов (элементы скелета, кровеносные сосуды, опухоли и т. д.), так и пустот, например внутреннюю поверхность лёгких. Задавая размер ячеек, можно получать либо более грубую триангуляцию, но содержащую меньшее количество треугольников, либо более точную, но со значительно ббльшим числом треугольников. Точность алгоритма трёхмерного роста области всегда зависит лишь от размера вокселя, который определяется только параметрами томографического обследования (размером пикселя и расстоянием между срезами). В построенной этим методом триангуляции число треугольников значительно больше, а само вычисление происходит медленнее, чем в ячеечных методах. Построенная триангуляция в ряде случаев нуждается в сглаживании.

С другой стороны, алгоритм роста области во многих ситуациях позволяет исключить из модели нежелательные объекты. Например, при детектировании внутреннего объёма лёгких мы автоматически исключаем пустоты внутри кишечника и трахеи, что при других способах (например, детектируя контуры лёгких на срезах) сделать очень трудно. Плюс этого алгоритма и в том, что он, в отличие от ячеечных методов, имеет простую реализацию, а построенная триангуляция гарантировано свободна от проблем (вырожденные треугольники, слишком острые углы и т. п.).

Для задачи построения модели аорты и коронарных артерий алгоритм роста области подходит лучше, поскольку в нём мы автоматически получаем триангуляцию внутренней поверхности кровеносных сосудов и объёма сердца, исключая, например, костную ткань, имеющую сходные значения плотности HU.

По качеству получаемых изображений эти два алгоритма в нашем случае отличаются мало. На рис. 5 приводятся два изображения аорты и части коронарных артерий, полученные разными алгоритмами.

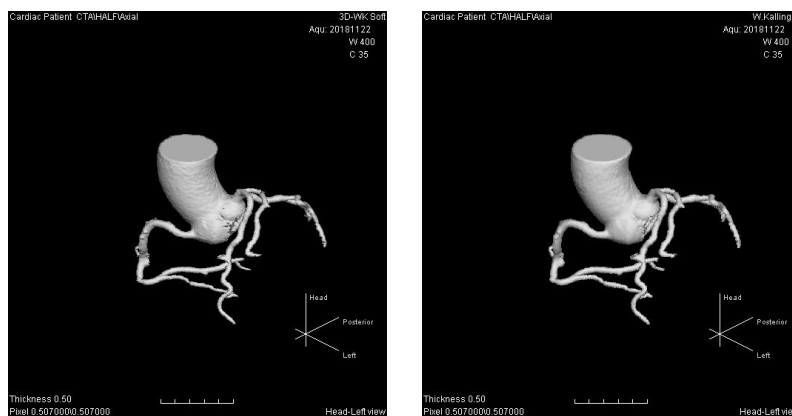


Рис. 5. Результаты трёхмерного восстановления аорты и части коронарных артерий. Левая модель получена с помощью ячеечного метода МТ6, правая трёхмерным алгоритмом роста области. Внешне изображения двух моделей отличаются незначительно, хотя алгоритмы принципиально разные

При рассмотрении «под микроскопом», однако, разница в результатах этих алгоритмов гораздо заметнее (рис. 6). Триангуляция, полученная вторым алгоритмом, состоит из поверхности кубиков (вокселей), во многих ситуациях требуется применить к ней алгоритм сглаживания.

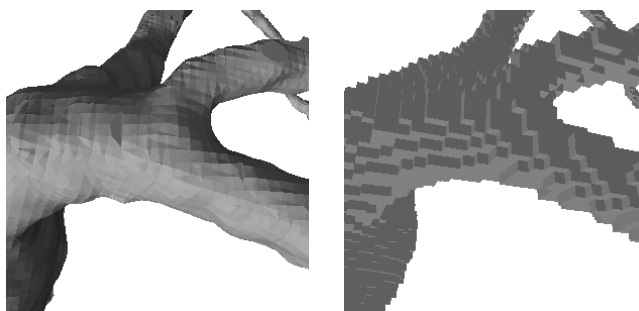


Рис. 6. Разница между ячеечным алгоритмом триангуляции и трёхмерным алгоритмом роста области. Крупно показан один и тот же фрагмент двух моделей, созданный разными алгоритмами; целиком они изображены на рис. 5

## Определение области интереса

В любых медицинских программных пакетах, предназначенных для работы с результатами КТ-исследования, есть возможность определения области интереса, которая кратко обозначается ROI (Region Of Interest); в трёхмерном случае используется также обозначение VOI (Volume Of Interest). Особенно важно точно определить VOI в случае трёхмерной реконструкции, поскольку всегда реконструкция выполняется не по всему объёму, а только для интересующего нас органа. Это может быть участок кости, кровеносного сосуда, бронхиальное дерево, внутренний объём лёгких и т. д. Большая часть трёхмерных алгоритмов восстанавливает объект по заданному пороговому значению HU, однако пороговое значение почти всегда не выделяет однозначно только интересующий нас объект. Например, в нашем случае требуется восстановить коронарные артерии и аорту. При добавлении в кровь контрастного вещества значения HU крови вырастают с 40–60 HU до нескольких сотен. Однако примерно те же значения плотности имеет, например, и костная ткань, кальцинаты и т. п.

Отделить сердце и кровеносную систему от костной ткани несложно, однако в нашем случае мы сталкиваемся с более серьёзной задачей: отделить вены и внутренний объём сердца от аорты и коронарных артерий. Сделать это только на основе значений HU невозможно, они близки для любых кровеносных сосудов. К сожалению, эту задачу очень трудно решить автоматически, здесь необходима ручная или полуавтоматическая работа человека, знающего анатомию хотя бы в минимальном объёме. На рис. 7 приведены результаты трёхмерного

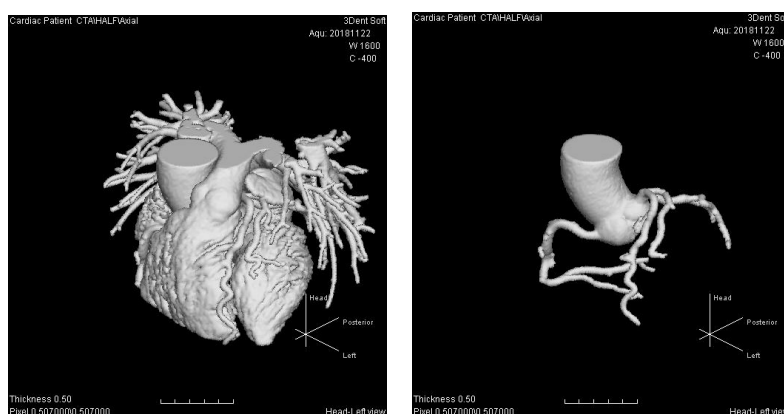


Рис. 7. Результаты трёхмерного восстановления области сердца с помощью трёхмерного алгоритма роста области. Левая модель получена с использованием простого задания VOI одним контуром на каждом срезе. Правая модель определяет VOI с помощью набора контуров на каждом срезе, эти контуры выделяют аорту и часть коронарных артерий из модели, изображённой слева

восстановления объёма сердца с использованием трёхмерного алгоритма роста области. Слева используется простейшее задание VOI в виде *одного контура* на каждом томографическом срезе. (Вручную каждый контур определять не обязательно, можно делать это, допустим, на каждом десятом срезе; на промежуточных срезах контуры вычисляются с помощью интерполяции.) Справа использовано полуавтоматическое определение *множества контуров* на каждом срезе, выделяющих области аорты и части коронарных артерий. По этим изображениям можно представить, насколько сложна задача выделения артерий.

### Полуавтоматическое задание VOI

При полуавтоматическом построении VOI для отделения аорты и коронарных артерий от внутреннего объёма сердца, вен и других артерий (см. рис. 7) используется следующая процедура. Сначала пользователь отмечает кликом мыши точку внутри аорты. (Возможна также реализация автоматического поиска точки в аорте.) Далее запускается трёхмерный алгоритм роста области, использующий в качестве семени отмеченную точку. Критерием принадлежности вокселя к области является значение плотности HU в этой точке, которое должно быть не ниже порогового значения; используется вариант алгоритма с шестью соседними вокселями. Пороговое значение является параметром алгоритма и зависит от типа и концентрации контрастного агента в крови; в приведённых выше примерах (рис. 7 и другие рисунки) использовалось пороговое значение  $HU = 180$ .

В результате работы алгоритма получается множество вокселей  $S_0$ , которое включает в себя внутренние объёмы аорты, сердца и связанных с ними кровеносных сосудов (коронарные артерии, другие артерии, вены; см. изображение слева на рис. 7). Множество вокселей  $S_0$  в пересечении с плоскостью каждого среза представляет собой уже набор двумерных связных областей, состоящих из множества пикселей. Для всех этих областей вычисляются ограничивающие их контуры, которые изображаются на каждом срезе.

### Алгоритм детектирования контура

Алгоритм построения контура для обхода связного множества пикселей близок по идее к алгоритму Мура трассировки окрестности (Moore Neighborhood Tracing [21]), в русскоязычной литературе такого рода алгоритм называют «алгоритмом жука» или «алгоритмом божьей коровки». В нашем случае отличие от классического алгоритма Мура состоит в том, что мы находим не граничные пиксели области, а непосредственно примыкающие к границе свободные пиксели. Идея алгоритма состоит в обходе препятствия в направлении против часовой стрелки, держась левой рукой за препятствие (рис. 8).

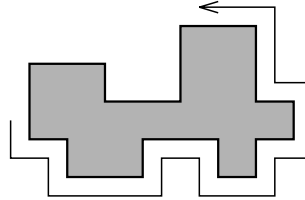


Рис. 8. Алгоритм построения контура вокруг связной области пикселей. Выполняется обход «препятствия» против часовой стрелки с касанием левой рукой его стенки

### Полуавтоматическая процедура выделения сосудов

После того как вычислено множество вокселей  $S_0$  (включающее весь внутренний объём сердца и кровеносных сосудов) и его визуальное представление в виде набора контуров на каждом срезе, начинается процесс полуавтоматического построения результирующего множества вокселей  $S_1 \subset S_0$  (множество  $S_1$  строится как подмножество  $S_0$ ); множество  $S_1$  должно включать лишь часть аорты и интересующие нас коронарные артерии. Для этого пользователь кликом мыши отмечает один из контуров — программа выбирает тот контур, *внутри* которого был сделан клик. Например, это может быть контур, ограничивающий верхнюю часть аорты. Все воксели на этом срезе, попадающие внутрь отмеченного контура, добавляются к множеству  $S_1$ . (Это в точности все воксели исходного множества  $S_0$ , по которым и был построен отмеченный контур.) После этого пользователь переходит к одному из томографических срезов, расположенных ниже, и точно так же выбирает второй контур; например, это может быть контур, ограничивающий нижнюю часть аорты. Программа в результате добавляет к множеству  $S_1$  все воксели внутри второго контура. После этого пользователь возвращается к одному из срезов, расположенных *между* первым и вторым срезами; например, это может быть срез, непосредственно предшествующий нижнему срезу или непосредственно следующий за верхним срезом (если считать, что срезы упорядочены в порядке от головы к ногам, именно так обычно работает компьютерный томограф).

Два контура на разных срезах отмечаются в режиме «Задание границ сосуда». После этого пользователь переключает режим на «Детектирование сосуда». Далее пользователь должен сделать клик внутри одного из промежуточных контуров, принадлежащих данному сосуду. После этого программа выполняет описанный выше трёхмерный алгоритм роста области из семени, используя в качестве начальной точки воксель, отмеченный пользователем. Воксели при этом добавляются к множеству  $S_1$ , причём в процессе работы рассматриваются лишь воксели, уже принадлежащие исходному множеству  $S_0$ .

Поскольку алгоритм роста области в процессе работы добавляет к области лишь свободные воксели, он не может пройти через границы, установленные пользователем в режиме задания границ (так как все воксели из граничных контуров уже добавлены к  $S_1$ ). Поэтому к множеству  $S_1$  будет добавлена лишь часть сосуда между отмеченными границами.

Части сосудов, добавленные к множеству  $S_1$  в процессе работы, выделяются цветом на изображениях срезов, так что пользователь постоянно контролирует процесс выделения нужных сосудов. Если алгоритм роста области выходит за желаемые границы (допустим, если пользователь не установил одну из границ, через разветвления сосудов и т. д.), то возможна откатка неудачного действия.

Важно отметить, что сосуд можно добавлять к множеству  $S_1$  по кусочкам. Отмечаемая граница вовсе не обязательно должна быть в самой верхней или самой нижней части сосуда. Можно отметить промежуточную границу, которая, например, сначала используется как нижняя граница детектируемого участка в верхней части сосуда, а затем как верхняя граница следующего участка, расположенного ниже по течению. Перед стадией детектирования в ряде случаев необходимо установить несколько границ; например, в случае разветвления сосуда нижняя граница должна быть установлена в обеих ветвях. В ряде случаев нижнюю границу можно не устанавливать, когда сосуд заканчивается и алгоритм роста области не проникает дальше него.

Помимо контуров, автоматически определяемых по множеству вокселей  $S_0$ , пользователь может добавлять контуры вручную, рисуя их мышью на срезах. Это может быть нужно, когда автоматический алгоритм роста области проникает в нежелательные участки. Контуры могут быть двух типов: отрицательные и положительные, соответственно они изображаются разными цветами. Нарисовав отрицательный контур, можно отделить интересующую область от нежелательной части: все воксели внутри отрицательного контура исключаются из процесса детектирования (логически мы как бы удаляем эти воксели из множества  $S_0$ , хотя в программе это реализовано несколько иначе). Наоборот, иногда нужно вручную добавить небольшое количество вокселей. Например, артерия может «исчезнуть» на одном из срезов и затем появиться вновь, это может быть связано как с её сужением (стеноз), так и с неравномерным распределением контрастного агента в крови или небольшим движением сердца в процессе сканирования. Пользователь может вручную нарисовать положительный контур, воксели внутри него будут (логически) добавлены к множеству  $S_0$ .

По окончании этой работы формируется множество вокселей  $S_1$ , включающее требуемую часть аорты и коронарных артерий. После этого выполняется построение триангуляции поверхности множества  $S_1$ .



## Сглаживание триангуляции

Триангуляция, полученная по воксельной трёхмерной модели, имеет негладкую форму (см. рис. 6 и 9). Это не столь существенно, когда нужно только получить изображение модели, поскольку использование нормалей и алгоритма Фонга (Phong Shading [22]) позволяет визуально сгладить изображаемую поверхность, тем более что размеры треугольников в нашем случае совсем небольшие (см. правое изображение на рис. 9). Но для гидродинамического расчёта форма границы уже важна, поскольку негладкая граница создаёт значительное сопротивление течению жидкости. Поэтому построенная триангуляция нуждается в сглаживании.

Известно несколько типов алгоритмов сглаживания поверхности, в применяемом программном пакете используется алгоритм из класса Subdivision Surface (дробление поверхности). На вход такому алгоритму подаётся трёхмерная модель поверхности из выпуклых многоугольников (не обязательно треугольников), на выходе он строит модель из более мелких многоугольников, которая сглаживает исходную модель. Алгоритм может применяться многократно, в пределе получается почти всюду  $C_2$ -гладкая поверхность, в отдельных точках  $C_1$ -гладкая. Количество шагов сглаживания (и измельчения многоугольников) можно задавать как параметр. Алгоритмов типа Subdivision Surface известно несколько, они отличаются типом используемых многоугольников (треугольники или четырёхугольники), а также тем, оставляет алгоритм неподвижными исходные вершины (схема *интерполяции*) или может их немного смещать (схема *аппроксимации*). В используемой программе применяется алгоритм Лупа [17], работающий с триангуляцией и использующий аппроксимирующую схему. Возможно также использование алгоритма Кетмулла—Кларка [11], работающего с моделью из четырёхугольников и также применяющего аппроксимирующую

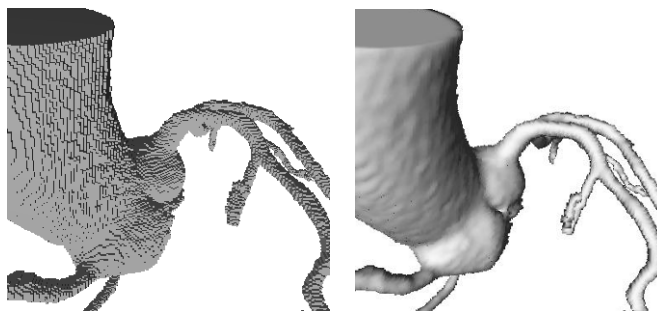


Рис. 9. Триангуляция поверхности множества вокселей, полученного трёхмерным алгоритмом роста области. На левом изображении грани вокселей закрашиваются как плоские многоугольники (flat shading). Правое изображение той же модели получено с помощью алгоритма Фонга (smooth shading): использование нормалей к поверхности, вычисленных как градиент функции плотности, позволяет визуально сгладить триангуляцию без изменения её формы

схему. В этом случае исходная модель является поверхностью вокселей, состоящей из прямоугольников; после сглаживания каждый четырёхугольник разбивается диагональю на два треугольника, чтобы в результате получить триангуляцию.

## Результаты экспериментов и заключение

Для построения трёхмерной модели аорты и коронарных артерий по результатам томографического обследования были использованы два алгоритма трёхмерного восстановления: ячеечный метод МТ6, основанный на разбиении пространства на тетраэдры, и трёхмерный алгоритм роста области из семени. Эксперименты проводились с помощью программного комплекса 3D-WK, представляющего собой универсальное программное обеспечение для работы с результатами томографии. В комплексе 3D-WK реализован большой набор типичных задач радиологии: обследование лёгких, измерение содержимого минералов в костной ткани, рабочая станция дантиста, исследование мозгового кровообращения и др. [7, 15, 16]. Реализация комплекса 3D-WK выполнена на языке C++ с использованием библиотеки классов MFC, а также свободного программного обеспечения для поддержки протоколов DICOM (Offis DCM Toolkit и GDCM Grassroots DICOM library). Полученные с его помощью трёхмерные модели подавались на вход программному комплексу для аэро- и гидродинамических расчётов FlowVision.

По результатам тестирования и частичной реализации необходимых полуавтоматических процедур можно сделать следующие выводы:

- использование трёхмерного алгоритма роста области из семени предпочтительнее, чем использование ячеечных алгоритмов построения триангуляции, поскольку алгоритм роста автоматически отслеживает только кровеносную систему, не включая посторонние детали;
- полуавтоматическую процедуру выделения аорты и коронарных артерий из общей модели кровеносной системы, полученной трёхмерным алгоритмом роста области, также удобно реализовать с помощью того же алгоритма. При этом пользователю нужно только отметить кликами мыши контуры, ограничивающие нужный участок кровеносного сосуда (контуры уже построены автоматически на первом этапе), и повторно запустить трёхмерный алгоритм роста области для этого участка. Результирующая модель строится как объединение выделенных таким образом участков;
- окончательно триангуляция строится как поверхность множества вокселей, выделенного из общей модели кровеносной системы в ходе интерактивной полуавтоматической процедуры;
- построенная поверхность нуждается в сглаживании, для этого можно применить один из алгоритмов сглаживания типа Subdivision Surface (дробление поверхности): алгоритм Лупа или алгоритм Кетмулла—Кларка.

## Литература

- [1] Аксенов А. А. FlowVision: индустриальная вычислительная гидродинамика // Компьют. исслед. и моделирование. — 2017. — Т. 9, № 1. — С. 5—20.
- [2] Аксенов А. А., Гудзовский А. В. Пакет прикладных программ FlowVision. — Тр. МФТИ. Сер. Аэрофизика и прикл. матем. — 1998. — С. 45—56.
- [3] Бончковский Р. Н. Заполнение пространства тетраэдрами // Матем. просв. Сер. 1. — 1935. — № 4. — С. 26—40.
- [4] Веселова Т. Н., Меркулова И. Н., Барышева Н. А., Терновой С. К., Шария М. А., Руда М. Я. Сравнение особенностей атеросклеротических бляшек в коронарных артериях у больных с острым коронарным синдромом и стабильной формой ишемической болезни сердца по данным мультиспиральной компьютерной томографии // Кардиология. — 2013. — Т. 53, № 12. — С. 14—20.
- [5] Веселова Т. Н., Меркулова И. Н., Яровая Е. Б., Терновой С. К., Руда М. Я. Роль мультиспиральной компьютерной томографии в оценке жизнеспособности миокарда и прогнозировании развития ремоделирования левого желудочка сердца у больных инфарктом миокарда с подъемом сегмента ST на ЭКГ // Кардиология. — 2013. — Т. 53, № 2. — С. 10—18.
- [6] Винберг Э. Б. Калейдоскопы и группы отражений. — Матем. просв. — 2003. — № 7. — С. 45—63.
- [7] Kallinger W., Борисенко В. В., Дижевский А. Ю. Распознавание шаровидных особенностей по серии томографических снимков // Тр. 18-й Междунар. конф. по компьют. графике и зрению Графикон'2008. GraphiCon. — М., 2008. — С. 312.
- [8] Кокстер Г. С. М. Введение в геометрию. — М.: Наука, 1966.
- [9] Терновой С. К., Шабанова М. С., Гаман С. А., Меркулова И. Н., Шария М. А. Роль компьютерной томографии в выявлении нестабильных атеросклеротических бляшек коронарных артерий: сопоставление результатов компьютерной томографии и внутрисосудистого ультразвукового исследования // Рос. электр. журн. луч. диагн. — <http://rejz.ru/>. — 2016. — Т. 6, № 3. — С. 68—79.
- [10] Carneiro B. P., Kaufman A. E. Tetra-Cubes: An algorithm to generate 3D isosurfaces based upon tetrahedra // SIGGRAPH'96. — P. 205—210.
- [11] Catmull E., Clark J. Recursively generated B-spline surfaces on arbitrary topological meshes // Computer-Aided Design. — 1978. — Vol. 10, no. 6. — P. 350—355.
- [12] Chernyaev E. V. Marching Cubes 33: Construction of Topologically Correct Isosurfaces. — Protvino: Inst. for High Energy Phys., 1995.
- [13] Gueziec A. Exploiting triangulated surface extraction using tetrahedral decomposition // IEEE Trans. Visualization Comput. Graphics. — 1995. — Vol. 1, no. 4. — P. 328—342.
- [14] Justice R. K., Stokely E. M., Strobel J. S., Ideker R. E., Smith W. M. Medical image segmentation using 3D seeded region growing // Proc. SPIE 3034. Medical Imaging 1997: Image Processing. — 1997.
- [15] Kallinger W., Fischer R., Borisenko V., Porteder H., Santha G., Springer E. Dental planning and patient safety // Wiener Klinische Wochenschrift. — 2008. — Vol. 120 (suppl. 3). — P. 52—54.

- [16] Kallinger W., Fischer R., Borisenko V., Santha G., Springer E. Computer aided detection of lung nodules in the view of patient safety // Wiener Klinische Wochenschrift. — 2008. — Vol. 120 (suppl. 3). — P. 49–52.
- [17] Loop Ch. Smooth Subdivision Surfaces Based on Triangles: M. S. Math. Thesis. — Univ. of Utah, 1987.
- [18] Lorensen W. E., Cline H. E. Marching Cubes: A high resolution 3D surface construction algorithm // Computer Graphics. — 1987. — Vol. 21, no. 4. — P. 163–169.
- [19] Serra J. Image Analysis and Mathematical Morphology. — Academic Press, 1982.
- [20] Skala V. Precision of isosurface extraction from volume data and visualization // Conf. on Sci. Comput. 2000. — P. 368–378.
- [21] Pavlidis G. Algorithms for Graphics and Image Processing. — Rockville: Comput. Sci. Press, 1982.
- [22] Bui Tuong Phong. Illumination of Computer-Generated Images. — UTEC-CSs-73-129. — Univ. of Utah, 1973.