

УДК 004.932.4:004.421

## ПРИМЕНЕНИЕ МЕХАНИЗМОВ ЭВРИСТИЧЕСКОГО ПОИСКА ДЛЯ ФОРМИРОВАНИЯ ЦЕПОЧЕК АЛГОРИТМОВ, УЛУЧШАЮЩИХ ГРАДИЕНТНЫЕ ИЗОБРАЖЕНИЯ

**В. В. Белов**, д.т.н., профессор кафедры ВПМ, РГРТУ; vvbeloff@yandex.ru

**А. К. Лопатин**, аспирант ГОУ ВО МО «ГСГУ»; ak\_lopatin@mail.ru

*Рассматривается задача формирования градиентных изображений изделий, находящихся на конвейерной ленте, в процессе реализации автоматического контроля их геометрических размеров. Целью работы является разработка методики автоматического синтеза алгоритма получения градиентных изображений надлежащего качества. Алгоритм синтезируется в виде цепочки необходимых процедур (операторов), используемых в практике решения задач обработки изображений. Структурная и параметрическая идентификация синтезируемой цепочки осуществляется автоматически с помощью модифицированного генетического алгоритма. Входными параметрами синтезируемой цепочки служат: изображение контролируемого изделия в RGB формате и бинарное изображение-эталон. Синтезируемая цепочка адаптируется к условиям съёмки (тип камеры, её разрешение, освещённость поверхности), формируемое ею изображение предназначается для последующего выделения краёв изделия с целью оценки его геометрических параметров. В процессе синтеза цепочки формируется вектор её операторов и их параметров.*

**Ключевые слова:** генетические алгоритмы, обработка изображений, композиции алгоритмов, оптический контроль, множество алгоритмов, фильтрация изображений, выделение краёв.

**DOI:** 10.21667/1995-4565-2018-66-4-1-78-89

### Введение

К настоящему времени создано большое количество алгоритмов решения задач, так или иначе связанных с обработкой изображений, но все они имеют одно общее начало: вся последовательность шагов этих алгоритмов – от начальной до конечной процедуры – является продуктом естественного (человеческого) интеллекта. Сложность решаемых задач, многообразие и изменчивость условий наблюдения проблемных объектов все более настойчиво стимулируют потребность в ускоренном, а, следовательно, *автоматическом* синтезе алгоритмов обработки изображений, включая определение состава необходимых процедур и их параметров.

В настоящей статье описаны результаты исследований, стимулом к проведению которых явилась потребность одного из промышленных предприятий, занятого производством деталей железнодорожного назначения, в реализации *автоматического* измерения габаритных размеров изделий, находящихся на конвейерной ленте, по их фотографическим снимкам. Была поставлена задача создания системы, способной обеспечить автоматическую адаптацию процесса измерения к колебаниям условий наблюдения

объектов в условиях реального производственного процесса. В рамках указанной задачи, в частности, была создана излагаемая методика формирования градиентных изображений, применяемых в дальнейшем для оценки качества изделий – верности их геометрических размеров и отсутствия неприемлемых повреждений поверхности.

Разработке методики предшествовала формулировка типичного для программных систем проектного требования: синтезируемая цепочка операторов должна быть оптимальной по двум критериям: она должна сочетать высокую точность восстановления контура изделия с минимумом трудоемкости.

### Анализ опыта автоматического синтеза алгоритмов обработки изображений

Вопросы автоматизации синтеза алгоритмов обработки изображений в публикациях представлены скудно. Применяемые в настоящее время подходы к автоматизации улучшения качества изображений, в частности, изложены в [2-7]. Можно выделить два основных подхода:

- 1) применение генетических алгоритмов [2-4];
- 2) применение искусственных нейронных сетей (ИНС) [6].

**Опыт синтеза цепочек стандартных алгоритмов в практике улучшения изображений**

В [2-4] генетический алгоритм используется для автоматизации процесса синтеза алгоритма обработки изображений, полученных в результате ультразвукового зондирования. Искомый алгоритм синтезируется генетическим алгоритмом в виде композиции алгоритмов, реализующих некоторые базовые процедуры (операторы). Целью синтеза является получение цепочки алгоритмов, обеспечивающей улучшение изображения до достижения максимального приближения к изображению эталона объекта, представленному на изображении.

Степень близости изображений оценивается значением фитнес-функции, вычисляемой в процессе *попиксельного* сравнения. Утверждается, что фитнес-функция может иметь различные определения, но конкретно она определяется однажды [3] с. 64 следующим образом:

$$Fitness = \min \left( \frac{|Q \cap Q'|}{|Q \cup Q'|}, \frac{|P \cap P'|}{|P \cup P'|} \right),$$

где  $Q, Q'$  – множества ярких пикселей,  $P, P'$  – множества черных пикселей эталонного и сопоставляемого фрагментов изображений соответственно.

Генетический алгоритм применяется как метод подтверждения оптимальности последовательности обработки изображений, предлагаемой автором на основе интуитивных соображений. Проводится анализ нескольких комбинаций базовых алгоритмов, и в зависимости от значения функции приспособленности (фитнес-функции) на различных наборах входных данных делается вывод о преимуществе предлагаемой автором методики обработки. В [3] содержится важный, хотя и достаточно очевидный вывод [3] с. 78: «Для некоторого нового изображения, не задавая шаблон, нельзя произвольным образом выбрать последовательность, из ранее построенных ГА, и ожидать требуемого результата обработки» (здесь ГА – «генетическим алгоритмом»).

Применительно к задаче автоматического измерения изделий этот вывод означает, что композиция алгоритмов, достаточно хорошо улучшающая изображение одного изделия, не гарантирует столь же хорошего улучшения изображения другого изделия. Для каждого типа изделия (тип изделия называют шаблоном, эталоном или образцом) требуется собственная структурная и, что гораздо чаще, – параметрическая идентификация алгоритмов обработки изображения. Выявлять, какую именно компо-

зицию алгоритмов нужно использовать в текущий момент, можно двумя способами: 1) решением задачи распознавания типа изделия на изображении; 2) экзогенным указанием необходимой информации, т.е. включением типа наблюдаемого изделия в состав исходных данных алгоритма. Первый способ применим в любом случае, а второй – только тогда, когда типы наблюдаемых изделий меняются редко или разные изделия наблюдаются на разных конвейерных лентах.

Среди особенностей методики, предлагаемой в [3], отметим потенциальную возможность многократного применения одного и того же алгоритма к одному и тому же изображению и произвольное расположение (в общем случае – произвольные расположения) алгоритмов в формируемой цепочке. В [3] утверждается, что эта особенность позволяет получить необходимый результат – достаточно высокое сходство эталона и результата обработки изображения. Однако, как показывает практика, указанное утверждение справедливо не всегда.

В контексте рассматриваемой задачи можно указать следующие недостатки методики [3-4].

1. Неупорядоченность и потенциальная повторяемость алгоритмов в формируемых цепочках приводит к тому, что генетический алгоритм, осуществляя перебор возможных цепочек, формирует и оценивает варианты, содержащие многократные повторения противонаправленных по характеру воздействия на изображение операторов. Например, могут формироваться и повторяться цепочки: «сглаживание (фильтрация, интегрирование) – выделение границ (вычисление градиента, дифференцирование) – сглаживание – выделение границ». В задачах, в которых зрительное сходство результатов обработки изображения с их эталонными вариантами непринципиально, а значение имеет только точность выделения контуров, повторение фильтрации и выделения границ лишено смысла и приводит к неоправданному затягиванию процедуры поиска оптимальной цепочки алгоритмов. Затягивание может быть столь существенным, что возникает потребность во вмешательстве экспертов с целью ускорения формирования цепочек.

2. Ориентация на максимизацию сходства результата обработки изображения с эталоном может привести к удалению целевой информации: могут быть удалены нарушения формы, трещины и каверны, являющиеся предметом выявления во время автоматического контроля качества изделий.

3. Предварительная определённость значений параметров алгоритмов. Аргументом в пользу именно предопределённых параметров служат

опыт и экспертные мнения. Отчасти это обосновано и практическими экспериментами [9], но в то же время существуют причины, по которым могут потребоваться изменения параметров алгоритмов, например, радиуса окна фильтра или порога оператора выделения границ:

- изменение модели фиксирующей аппаратуры;
- изменение условий съёмки;
- микродефекты конкретного фотоаппарата.

Указанное означает, что в общем случае параметрическая идентификация алгоритмов является необходимой процедурой.

4. Достаточно существенное участие эксперта в процессе формирования оптимальной цепочки алгоритмов. Эта особенность является совершенно неприемлемой, если процесс формирования новой цепочки алгоритмов становится частью технологического процесса повседневной эксплуатации программного обеспечения.

#### ***Опыт улучшения качества изображений на основе функции пересчета яркости пикселей***

Идея метода [2], интенсивно использованная в многочисленных трудах авторов [3-5], заключается в повторяющемся пересчете текущих значений элементов матрицы яркостей пикселей изображения в новые ожидаемо лучшие значения. Для пересчета используется функция, называемая ядром улучшения, с некоторым набором параметров. В [2] количество параметров ядра улучшения равно четырем.

Значения параметров подбираются с помощью генетического алгоритма. По сути дела, пересчет матрицы яркостей пикселей реализуется как составная часть решения задачи оптимизации, в которой целевой функцией является качество изображения, а искомыми переменными – параметры ядра улучшения. Повтор пересчетов яркостей пикселей осуществляется до выполнения критерия останова генетического алгоритма – крайне редко завершение осуществляется по причине достижения глобального, либо субоптимального решения, гораздо чаще останов происходит из-за исчерпания числа поколений, отпущенных на эволюцию, либо исчерпания времени, отпущенного на эволюцию.

Целевая функция представляет собой свертку трех частных критериев. В [2] предложено три частных критерия: 1) количество граничных пикселей; 2) интенсивность краевых пикселей; 3) мера энтропии изображения. Для вычисления первых двух критериев используется дискретный дифференциальный оператор Собела, вычисляющий приближённое значение градиента яркости изображения. Авторы [3-5] добавили к указанным

трем частным критериям еще один: уровень адаптации к зрению человека по яркости.

Таким образом, в алгоритмах, развивающих идею метода [2], генетические алгоритмы используются в качестве альтернативы стандартным процедурам улучшения качества изображений, имеющим четкое математическое описание. Хотя полного отказа от стандартных процедур не происходит: 1) для вычисления двух частных критериев используется стандартный оператор вычисления градиента яркости изображения (конечно же, можно использовать не только оператор Собела [8], но и любой другой); 2) некоторые модификации [5] предполагают двухэтапные решения – на первом этапе реализуется стандартная процедура улучшения изображения за счет выравнивания гистограммы, а на втором запускается алгоритм пересчета яркостей в контексте решения задачи оптимизации с помощью генетического алгоритма [2].

#### ***Предлагаемая методика синтеза цепочек алгоритмов, улучшающих качество градиентных изображений*** ***Рабочие гипотезы***

В процессе обобщения опыта автоматизации процессов структурной и параметрической идентификации алгоритмов обработки изображений были сформулированы следующие рабочие гипотезы.

1. Существует определенная «целесообразная» последовательность шагов (функций, частных задач) в процессе решения задачи улучшения градиентных изображений изделий, представленных на обрабатываемых снимках. Каждый шаг – это применение того или иного оператора (алгоритма) обработки изображения конкретной функциональности.

Эта последовательность шагов такова:

- 1) выравнивание гистограммы изображения;
- 2) фильтрация (сглаживание) изображения;
- 3) выделение границ путем вычисления градиента, т.е. получение градиентного (контурного) изображения;

4) уточнение границ – процесс минимизации толщины контуров, имеющий многочисленные альтернативные способы реализации, простейший из которых представляет собой совокупность следующих процедур:

4.1) закрытие – последовательное применение дилатации (расширения) и эрозии (разъедания), приводящее к удалению в контурах разрывов длиной до двух пикселей;

4.2) удаление одиночных пикселей – пиксели равные 1, все 8 соседей которых равны 0, заменяются на 0.

Использование конкретного способа уточнения границ принципиального значения не имеет.

2. Повторное применение любого из указанных шагов, кроме фильтрации, нецелесообразно, т. е. не способствует повышению точности восстановления контура изделия. Более того, повторное применение операторов выделения границ является деструктивным, – в частности, приводит к появлению повторений контуров.

3. Естественный тренд: чем сложнее алгоритм, тем более высокое качество обработки изображений он обеспечивает (в частности, операторы выделения границ с большей трудоемкостью обеспечивают более точное восстановление контуров объектов по их фотографиям) при использовании цепочек алгоритмов проявляется слабо, – практически отсутствует.

4. При решении конкретной прикладной задачи шаги обработки изображений могут быть поделены на обязательные и необязательные. В частности, в процессе улучшения качества градиентных изображений обязательным является шаг выделения границ, все остальные шаги факультативны, т.е. могут не выполняться (при высоком качестве фотографий).

Указанные гипотезы ни разу не были опровергнуты в процессе многочисленных экспериментов с изображениями изделий, подлежащих контролю во время их движения на конвейерной ленте. На основе этих гипотез разработана методика синтеза цепочек алгоритмов, улучшающих качество градиентных изображений. Методика позволяет найти цепочку, оптимальную по трудоемкости в ограничениях на точность восстановления контуров объектов, представленных на обрабатываемых снимках. В качестве показателя точности восстановления контуров используется значение линейного коэффициента корреляции между результатом обработки изображения цепочкой алгоритмов и соответствующим изображением-шаблоном.

#### **Укрупненные этапы методики**

Методика состоит из следующих укрупненных этапов.

1. Определение / пополнение операций – групп функциональных *операторов* (алгоритмов), из которых формируются цепочки: для каждого шага решения поставленной задачи определяются наборы альтернативных операторов. Очень важен следующий факт: в состав каждой группы обязательно включается «пустой» оператор, – не имеющий программного кода. «Включение» этого оператора в цепочку означает, что ни один из операторов данной группы не используется, т.е. соответствующая

функциональная операция в формируемом алгоритме не предусматривается.

Начальным набором правил, позволяющим наполнять пул алгоритмов, служит система аксиом Цермело-Френкеля. Если предлагаемый алгоритм при добавлении его к существующей системе нарушает эту систему аксиом, то предлагаемый алгоритм отбрасывается.

2. Определение *целесообразной последовательности* операторов, определяющей структуру возможных цепочек, образующих алгоритм решения поставленной задачи:

- а) выравнивание гистограммы изображения;
- б) фильтрация;
- в) выделение границ;
- г) уточнение границ.

3. Задание значения показателя, характеризующего точность восстановления контура изделия.

4. Осуществление поиска оптимальной цепочки с помощью специального генетического алгоритма, в качестве фитнес-функции которого используется показатель точности восстановления контура изделия.

В общем случае для реализации генетического алгоритма могут потребоваться значительные временные затраты, поэтому он имеет специальный экзогенно задаваемый параметр  $\Lambda$  – количество попыток найти лучшую цепочку.

5. Осуществляется принятие решения по результатам работы генетического алгоритма: выбирается «лучшая» цепочка из множества Парето, сформированного генетическим алгоритмом.

6. Формируется отчет о том, – найдена ли цепочка операторов, удовлетворяющая предъявленным точностным требованиям, и если найдена, то приводятся сведения о структуре цепочки, параметрах операторов, входящих в эту цепочку, и показателях точности и трудоемкости, обеспечиваемых этой цепочкой. Если цепочка, обеспечивающая требуемую точность восстановления контуров, не найдена, то констатируется факт невозможности решения поставленной задачи по причине недостаточности состава операторов, определенных на первом этапе, и необходимости возврата к первому этапу с целью пополнения состава функциональных операторов.

Указанная последовательность этапов образует методический (укрупненный) основной поток действий. Алгоритмические основной и альтернативные потоки определяются в процессе декомпозиции укрупненных этапов.

#### **Специфика генетического алгоритма, применяемого для поиска цепочки операторов**

Во-первых, особенностью алгоритма поиска является то, что он осуществляет одновременно

и параметрическую, и структурную идентификацию формируемых цепочек.

Во-вторых, отличительной его особенностью является следующее:

а) в процедуре каждой мутации при формировании очередной цепочки операторов жестко фиксируется последовательность операций, мутируются только операторы, реализующие операцию;

б) операция выравнивания гистограммы изображения может выполняться только один раз;

в) операция фильтрации может повторяться, при этом допускается как использование одного и того же оператора, так и применение различных операторов фильтрации;

г) операция выделения границ может выполняться только один раз;

д) операция уточнения границ может выполняться только один раз. Все операции, кроме выравнивания гистограммы, являются обязательными.

В-третьих, спецификой применяемого варианта генетического алгоритма является то, что в процессе его работы:

а) оценивается не только фитнес-функция (показатель точности восстановления контура изделия), но и показатель трудоемкости формируемых цепочек операторов;

б) формируется множество Парето с ограничением, – содержащее сведения о цепочках, удовлетворяющих требованию к точности восстановления контура изделия, и, одновременно, не доминирующих друг друга по точности и трудоемкости.

Выполнив формирование очередного поколения цепочек, главный поток генетического алгоритма передает управление потоку, реализующему цепочки сформированного поколения, который не только выполняет операторы цепочек, но и оценивает результаты их работы – точности восстановления контура и длительности работы всех цепочек. Эти результаты передаются в поток, который ведет упорядоченное множество Парето с ограничением, используя утверждения следующих теорем.

**Теорема 1.** Множество Парето с двумя критериями качества элементов не может содержать элементы с одинаковыми значениями однопозиционных критериев.

**Доказательство.** Пусть дано некоторое множество Парето:

$$P = \{p(a_1, b_1), p(a_2, b_2), \dots, p(a_n, b_n)\},$$

с элементами, характеризуемыми двумя критериями  $a$  и  $b$ .

Согласно определению, во множестве не может быть двух одинаковых элементов (эле-

ментов с одинаковыми значениями критериев), т.е. справедливо утверждение:

$$(\neg \exists i, j \in \{1, 2, \dots, n\}, i \neq j)(a_i = a_j) \& (b_i = b_j).$$

Кроме этого, справедливо утверждение:

$$(\neg \exists i, j \in \{1, 2, \dots, n\}, i \neq j)(a_i = a_j) \& (b_i \neq b_j),$$

поскольку либо  $b_i < b_j$ , либо  $b_i > b_j$ , то либо  $i$ -й, либо  $j$ -й элемент окажется доминированным по второму критерию и подлежащим удалению из множества Парето.

Аналогично справедливо утверждение:

$$(\neg \exists i, j \in \{1, 2, \dots, n\}, i \neq j)(a_i \neq a_j) \& (b_i = b_j),$$

поскольку либо  $a_i < a_j$ , либо  $a_i > a_j$ , то либо  $i$ -й, либо  $j$ -й элемент окажется доминированным по первому критерию и подлежащим удалению из множества Парето.

Из справедливости указанных выше утверждений следует справедливость утверждения:

$$(\neg \exists i, j \in \{1, 2, \dots, n\}, i \neq j)(a_i = a_j) \vee (b_i = b_j),$$

или в альтернативной формулировке:

$$(\forall i, j \in \{1, 2, \dots, n\}, i \neq j)(a_i \neq a_j) \& (b_i \neq b_j).$$

**Теорема 2.** Сортировка элементов множества Парето с двумя критериями качества элементов по одному из критериев автоматически приводит к обратной сортировке по второму критерию, если критерии имеют одинаковое качественное направление, и к прямой сортировке по второму критерию, если критерии качественно разнонаправлены.

**Доказательство.** Пусть дано некоторое множество Парето:

$$P = \{p(a_1, b_1), p(a_2, b_2), \dots, p(a_n, b_n)\}$$

с элементами, характеризуемыми двумя критериями  $a$  и  $b$ .

В силу контекстной симметричности параметров достаточно рассмотреть случай сортировки по первому параметру. В результате упорядочения по убыванию множества  $P$  по первому параметру  $a$  формируется вектор:

$$\vec{P} = (p(a_{i_1}, b_{i_1}), (a_{i_2}, b_{i_2}), \dots, (a_{i_n}, b_{i_n})),$$

в котором, согласно семантике упорядочения и первой теореме, справедливо следующее:

$$a_{i_j} > a_{i_{j+1}}, b_{i_j} \neq b_{i_{j+1}}, j \in \{1, 2, \dots, n-1\}.$$

Одновременно с этим случай  $b_{i_j} > b_{i_{j+1}}$  невозможен, поскольку  $i_{j+1}$ -й элемент оказывается доминированным, следовательно, возможен только следующий вариант:  $b_{i_j} < b_{i_{j+1}}$ .

Итак, для множества Парето с двумя критериями качества элементов, отсортированного по убыванию первого критерия, при одинаковом

качественном направлении критериев справедливо утверждение:

$$(\forall j \in \{1, 2, \dots, n-1\})(a_{i_j} > a_{i_{j+1}}) \& (b_{i_j} < b_{i_{j+1}}).$$

Теорема доказана.

Поток множества Парето содержит множество в отсортированном состоянии, что позволяет быстро определять потенциальное место новой цепочки в этом множестве, выяснять вопрос о вхождении либо не вхождении новой цепочки во множество Парето и при необходимости удалять имеющиеся цепочки, доминируемые новой цепочкой. Естественно, прежде всего, выясняется вопрос о допустимости новой цепочки, – факт выполнения требования к точности восстановления контура изделия.

По завершении работы основного потока генетического алгоритма управление передается методу принятия решения о «лучшей» цепочке из сформированного множества Парето.

#### **Правило выбора лучшей цепочки**

Сформированное множество Парето состоит из допустимых по точности цепочек. В рамках текущих условий решения задачи критерий трудоемкости является второстепенным, на него не накладываются ограничения, и его минимизация не требуется. Этот критерий учитывается в соответствии с принципом, отражающим потенциальную возможность возникновения требований ко времени работы программы в будущем: трудоемкость применяемого алгоритма должна быть минимальной в пространстве допустимых решений. С учетом указанных фактов в качестве лучшей цепочки следовало бы использовать последний элемент в отсортированном множестве Парето – цепочку с допустимой точностью восстановления контура и минимальным временем ее работы. Это соответствует классическому подходу минимизации одного из критериев с включением остальных критериев во множество ограничений.

Более рациональным представляется предлагаемый подход, основанный на использовании понятий нечеткой оптимизации и критерия максимума относительного преимущества.

Назовем нечеткой минимизацией требования к используемым критериям «быть по возможности» максимальным / минимальным, естественно, при соблюдении всех ограничений, предъявляемых, в том числе, и к используемым критериям.

Относительное преимущество  $i$ -го варианта при использовании качественно разнонаправленных критериев определим следующим образом:

$$B_i = 1; \\ B_i = \left(1 - \frac{K_i^{[2]}}{K_{\max}^{[2]}}\right) / \left(1 - \frac{K_i^{[1]}}{K_{\max}^{[1]}}\right), i = 2, 3, \dots, n,$$

где  $K_i^{[1]}$ ,  $K_i^{[2]}$  – соответственно значения первого (качественно положительного) и второго (качественно отрицательного) критериев для  $i$ -го варианта;  $K_{\max}^{[1]}$ ,  $K_{\max}^{[2]}$  – соответственно максимальные значения первого и второго критериев среди всех вариантов. Поскольку варианты предполагаются отсортированными по убыванию первого критерия, то справедливы равенства:  $K_{\max}^{[1]} = K_1^{[1]}$ ,  $K_{\max}^{[2]} = K_1^{[2]}$ .

Относительное преимущество имеет следующую семантику: оно показывает, во сколько раз относительный выигрыш

$$win = \frac{K_{\max}^{[2]} - K_i^{[2]}}{K_{\max}^{[2]}},$$

по второму критерию превышает относительный проигрыш

$$loss = \frac{K_{\max}^{[1]} - K_i^{[1]}}{K_{\max}^{[1]}},$$

по первому. При одинаковой структуре указанных дробей первая выражает выигрыш, потому что второй критерий качественно отрицателен, а вторая дробь выражает проигрыш, поскольку первый критерий качественно положителен. Относительное преимущество выражает некоторый рациональный компромисс между двумя критериями, используемыми для выбора лучшего варианта.

Предлагаемое правило выбора лучшей цепочки имеет вид: в качестве лучшей цепочки алгоритмов по критериям максимума точности восстановления контура и минимума времени её работы следует использовать цепочку, входящую в множество Парето и имеющую максимум относительного преимущества в этом множестве. Значение качественно положительного критерия, соответствующее максимуму относительного преимущества, назовем рациональным максимумом, соответствующее значение качественно отрицательного критерия назовем рациональным минимумом.

#### **Вычисление показателя точности восстановления контуров изображения**

Обозначим  $\mathbf{P}$  множество координат пикселей рассматриваемых изображений, т.е.

$$\mathbf{P} = \{(x, y) : x \in \{0, 1, \dots, M\}, y \in \{0, 1, \dots, N\}\}$$

где  $M, N$  – количество пикселей в изображении по вертикали и горизонтали соответственно.

Назовем функцией изображения функцию  $f(x, y)$ , отображающую элементы  $\mathbf{P}$  в значения интенсивности свечения (яркости) пикселей изображения:  $f : \mathbf{P} \rightarrow \mathbf{R}^n$ , где  $\mathbf{R}$  – множество вещественных чисел,  $n$  – целое число, опреде-

ляемое форматом представления цветности изображения. Например,  $n = 3$  для формата RGB,  $n = 1$  для черно-белых изображений. Дальнейшие рассмотрения ограничим именно черно-белыми изображениями. Обратим внимание на факт: при фотосъемке значения функции изображения формируются посредством оптикоэлектронных процессов.

Определим формально изображение как матрицу значений функции изображения:

$$\mathbf{F} = [f(x, y) : x \in (0, 1, \dots, M), y \in (0, 1, \dots, N)].$$

Замечание: в литературе, следуя [1], упрощенно без уточняющей формализации изображениями называют функции изображения, т.е. говорят так: «рассмотрим изображения  $f(x, y)$  и  $g(x, y)$ ».

Если изображение  $\mathbf{F}$  подвергается обработке алгоритмом  $\mathbf{A}$  и в результате формируется новое изображение  $\mathbf{U} = \mathbf{A}(\mathbf{F})$ , то можно записать:

$$\mathbf{U} = [u(x, y) : x \in (0, 1, \dots, M), y \in (0, 1, \dots, N)],$$

где  $u : \mathbf{P} \rightarrow \mathbf{R}$  – новая функция изображения.

Если имеется некоторое дополнительное изображение

$$\mathbf{V} = [v(x, y) : x \in (0, 1, \dots, M), y \in (0, 1, \dots, N)],$$

то можно вычислить линейный коэффициент корреляции  $r_{u,v}$  изображений  $\mathbf{U}$  и  $\mathbf{V}$ , определяемый формулой:

$$r_{u,v} = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [u(x, y) - \bar{u}][v(x, y) - \bar{v}]}{\sqrt{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [u(x, y) - \bar{u}]^2 \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [v(x, y) - \bar{v}]^2}}, \quad (1)$$

где  $u(x, y)$ ,  $v(x, y)$  – это значения функций изображений, т.е. яркости пикселей с координатами  $x, y$ ;  $\bar{u}, \bar{v}$  – средние значения яркостей пикселей изображений.

В изложенной методике синтеза цепочек алгоритмов, улучшающих качество градиентных изображений, линейный коэффициент корреляции  $r_{u,v}$  между градиентным изображением  $u(x, y)$ , полученным в результате обработки снимка изделия  $i$ -й цепочкой алгоритмов, и изображением-шаблоном  $v(x, y)$ , рассматриваемым в качестве идеального результата обработки того же снимка, используется в качестве показателя точности восстановления контура изображения. Коэффициент  $r_{u,v}$  выражает степень сходства изображений  $u(x, y)$  и  $v(x, y)$ , поэтому он является качественно положительным критерием  $K_i^{[1]}$  в процессе поиска лучшей цепочки, т.е.  $K_i^{[1]} = r_{u,v}$ .

### Вычисление показателей трудоемкости функциональных операторов и их цепочек

Оценивание показателей трудоемкости функциональных операторов по классической методике [11, 12] экспериментального исследования, основанной на подсчете числа выполняемых операций при различных наборах входных данных, практически невозможно из-за высокой трудоемкости процесса оценивания и отсутствия исходных кодов при использовании сторонних программ. По этой причине предлагается оценивать трудоемкость по временным затратам.

В качестве показателя  $d_i$  трудоемкости  $i$ -й цепочки операторов используется сумма значений показателей трудоемкости функциональных операторов, вошедших в цепочку  $d_i = \sum_{j=1}^n d_{j_i}^{[j]}$ ,

$i = \overline{1; m}$ , где  $n$  – количество функциональных групп операторов, из которых формируются цепочки;  $m$  – количество сформированных цепочек;  $d_{j_i}^{[j]}$  – показатель трудоемкости оператора  $j$ -й функциональной группы, вошедшего в  $i$ -ю цепочку. Показатель  $d_i$  трудоемкости  $i$ -й цепочки операторов является качественно отрицательным критерием  $K_i^{[2]}$  в процессе поиска лучшей цепочки, т.е.  $K_i^{[2]} = d_i$ .

### Альтернативный показатель сложности функциональных операторов и их цепочек

Показатель трудоемкости алгоритма, определяющий затраты процессорного времени, является одним из компонентов так называемого показателя ресурсоемкости. По этой причине более содержательным показателем сложности алгоритма является функция, интегрирующая показатель трудоемкости с показателем объема памяти, занимаемого этим алгоритмом, например, классическая линейная форма. В частности, сопоставлять операторы в рамках одной функциональной группы можно с помощью следующего показателя:

$$c_i^{[j]} = \alpha \frac{d_i^{[j]}}{d_{\max}^{[j]}} + \beta \frac{\mu_i^{[j]}}{\mu_{\max}^{[j]}}, \quad i = \overline{1; n_j},$$

где  $c_i^{[j]}$  – показатель ресурсоемкости  $i$ -го оператора  $j$ -й функциональной группы;  $n_j$  – количество операторов в  $j$ -й функциональной группе;  $d_{\max}^{[j]}$  – максимальная трудоемкость среди трудоемкостей операторов  $j$ -й функциональной группы, т.е.  $d_{\max}^{[j]} = \max(d_1^{[j]}, d_2^{[j]}, \dots, d_{n_j}^{[j]})$ ;  $\mu_i^{[j]}$  – объем памяти, занимаемый программой, реали-

зующей  $i$ -й оператор  $j$ -й функциональной группы;  $\mu_{\max}^{[j]} = \max(\mu_1^{[j]}, \mu_2^{[j]}, \dots, \mu_{n_j}^{[j]})$ ;  $\alpha, \beta$  – весовые коэффициенты, определяемые экспертным путем, удовлетворяющие условию:  $\alpha + \beta = 1$ .

Для сравнения ресурсоемкостей цепочек операторов можно использовать следующий показатель:

$$C_i = \alpha \frac{d_i}{d_{\max}} + \beta \frac{\mu_i}{\mu_{\max}}, i = \overline{1; m}, \quad (2)$$

где  $C_i$  показатель ресурсоемкости  $i$ -й цепочки операторов;  $d_{\max} = \max(d_1, d_2, \dots, d_m)$  – максимальная трудоемкость среди трудоемкостей всех сформированных цепочек;  $\mu_i = \sum_{j=1}^n \mu_{j_i}^{[j]}$  – объем памяти, занимаемый программой, реализующей  $i$ -ю цепочку операторов;  $\mu_{j_i}^{[j]}$  – объем памяти, занимаемый программой, реализующей оператор  $j$ -й функциональной группы, вошедший в  $i$ -ю цепочку.

Очень важен факт: показатель  $C_i$  ресурсоемкости  $i$ -й цепочки операторов не сводится к сумме показателей ресурсоемкостей операторов, входящих в эту цепочку, т.е.  $C_i \neq \sum_{j=1}^n C_{j_i}^{[j]}$ , где  $j_i$  – номер оператора  $j$ -й группы, входящего в  $i$ -ю цепочку, поскольку нормализация частных показателей (трудоемкостей и объемов памяти) осуществляется максимальными значениями сопоставляемых элементов разных множеств – отдельных групп операторов и множества всех сформированных цепочек.

Этот факт обусловлен тем, что в ходе выполнения цепочки алгоритмов на скорость выполнения операций оказывают влияние многие факторы, находящиеся вне контроля программиста, в частности: выравнивание данных, сборка мусора, параллелизм на уровне команд и т.п. В том числе возможна ситуация, при которой между конкретными алгоритмами различных функциональных групп не потребуется выполнять «промежуточные» неконтролируемые действия, а между другими – придётся.

Показатель ресурсоемкости  $i$ -й цепочки операторов  $C_i$  может служить альтернативным качественно отрицательным критерием  $K_i^{[2]}$  в процессе поиска лучшей цепочки, т.е.  $K_i^{[2]} = C_i$ .

### Иллюстративный пример применения предложенной методики для решения прикладной задачи

Для решения конкретной прикладной задачи по улучшению градиентных изображений изделий железнодорожного назначения в процессе контроля их габаритных размеров [10] оказалось достаточным использовать небольшое количество алгоритмов (см. таблицу). Этот набор позволяет устранить отклонения обрабатываемого изображения от эталона, вызванные различными причинами, в том числе:

- смазывание, вызванное движением конвейерной ленты;
- изменение освещённости;
- случайные выбитые пиксели на изображении, возникшие из-за дефекта матрицы фотокамеры (случайное попадание пыли, чистота материалов и качество подложки матрицы).

Применение порога в алгоритмах выделения границ позволяет считать пиксель относящимся к границе, если соответствующий ему пиксель результата фильтрации имеет значение, большее указанной величины. Особенностью реализации оператора Кэнни в среде Matlab является возможность применения порога, представляемого либо скалярной переменной, либо двухэлементным вектором. Для классификации перепадов на «слабые» и «сильные» используются два порога – нижний и верхний. «Слабые» границы отмечают в результирующем изображении, только если они соединены с «сильными». Если параметр *порог* является скалярным значением, то *порог* задает значение верхнего порога, а для нижнего порога используется значение  $0,4 * \text{порог}$ . Входной параметр, в зависимости от типа представления изображения может быть как целочисленным, так и вещественным. Предлагаемая модификация генетического алгоритма изначально настроена на работу с вещественными значениями.

Параметр *направление* используется только в операторе Собела и может принимать одно из трёх значений:

- '*horizontal*' – выделение горизонтальных границ;
- '*vertical*' – выделение вертикальных границ;
- '*both*' – выделение границ во всех направлениях.

Выделение границ только одного вида является необоснованным, поэтому в предлагаемой модификации генетического алгоритма используется поиск границ во всех направлениях.

## Применяемые алгоритмы обработки изображений, их параметры и особенности

| Алгоритмы фильтрации       |                                           |                                                                                                                                                                                          |
|----------------------------|-------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Алгоритм                   | Параметры                                 | Особенности                                                                                                                                                                              |
| «Пустой оператор»          | Отсутствуют                               | Не оказывается никакого воздействия на изображение. Включен в список для оптимизации времени работы алгоритма: применяется в случае, если исходное изображение не нуждается в фильтрации |
| Усредняющий фильтр         | Радиус                                    | Радиус задаётся двухкомпонентным вектором                                                                                                                                                |
| Фильтр Гаусса              | Радиус, среднее квадратическое отклонение | Параметр среднее квадратическое отклонение распределения Гаусса, которое используется при формировании маски                                                                             |
| Медианный фильтр           | Радиус                                    | Наиболее медленный алгоритм фильтрации. Фильтр не искажает резкие границы объекта                                                                                                        |
| Операторы выделения границ |                                           |                                                                                                                                                                                          |
| Оператор Собела            | Порог, направление                        | Пиксель считается относящимся к границе, если соответствующий ему пиксель результата фильтрации имеет значение, большее <i>порог</i>                                                     |
| Оператор Прюитт            | Порог                                     |                                                                                                                                                                                          |
| Оператор Кэнни             | Порог, среднее квадратическое отклонение  |                                                                                                                                                                                          |
| Лапласиан-гауссиана        | Порог, среднее квадратическое отклонение  | Выделение границ осуществляется после применения фильтра, аналогичного последовательному применению фильтров Гаусса и Лапласа                                                            |
| Оператор Робертса          | Порог                                     | Пиксель считается относящимся к границе, если соответствующий ему пиксель результата фильтрации имеет значение, превышающее <i>порог</i>                                                 |

Изображения, получаемые фотографированием изделий в производственных условиях, оказывались достаточно хорошими, и программа, реализующая предлагаемую методику, отыскивала цепочку операторов, обеспечивающих коэффициент корреляции между результатом обработки и шаблоном не менее  $r_{u,v}^{[min]} = 0,97$  при назначенном допустимом значении коэффициента  $r_{u,v}^{[доп]} = 0,95$ . Временные затраты при этом составляли от 1 до 100 мс.

Для наблюдения поведения методики в более сложных условиях проводился эксперимент с искусственно подпорченными изображениями – на получаемые фотографированием изображения накладывался десятипроцентный гауссов шум (путем установки соответствующего значения параметра «эффект» пакета Adobe Photoshop). Одно из тестовых изображений приведено на рисунке 1.

В ходе решения генетический алгоритм создаёт и нежизнеспособные особи, – со значениями параметров, не имеющими физического смысла либо выходящими за границы рациональных интервалов.

В процессе поиска наилучшей цепочки алгоритмов доля нежизнеспособных особей составляла от 15 до 18 %. Нежизнеспособные особи появлялись в каждом поколении популяции.

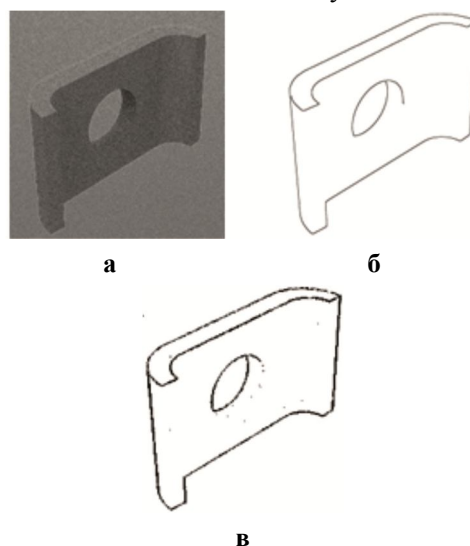


Рисунок 1 – а – тестовое изображение прижимной скобы; б – изображение-эталон (контуры выделены экспертом); в – результат обработки цепочкой, сгенерированной генетическим алгоритмом

Около половины жизнеспособных особей имело значение фитнес-функции (линейного коэффициента корреляции  $r_{u,v}$ ), превышающее 0,75. Показатели качества таких особей использовались для формирования статистических данных.

Для получения искомого результата поиск ограничивался семью поколениями. Получаемое финальное множество Парето имело мощность от 47 до 573. Цепочки, квалифицируемые как лучшие по критерию максимума относительного преимущества, имели значение линейного коэффициента корреляции (рационального максимума качественно положительного критерия  $K_i^{[1]} = r_{u,v}$ ) от 0,95 до 0,98. При этом значения рационального минимума качественно отрицательного критерия  $K_i^{[2]} = d_i$  (длительностей работы наилучших цепочек) находились в интервале от 8 до 110 мс.

Рисунком 2 наглядно иллюстрирован тот факт, что среди жизнеспособных особей, собранных со всех поколений генетического алгоритма, не наблюдается зависимости качества обработки изображения от быстродействия цепочки алгоритма. Фитнес-функция (ордината точки) практически равномерно распределена во всем интервале измерений. Длительность выполнения цепочки (абсцисса точки) алгоритмов не коррелирует с значением фитнес-функции.

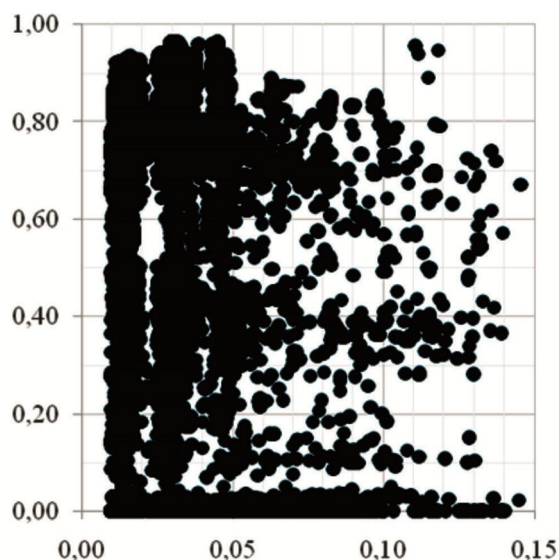


Рисунок 2 – Зависимость фитнес-функции от ресурсоёмкости цепочки алгоритма

Приведённый пример использования предлагаемой методики иллюстрирует подтверждение рабочих гипотез, использованных в процессе создания методики, среди которых наиболее заметной является гипотеза о практической независимости качества работы цепочки алгоритмов от ее трудоемкости.

Заметим следующее:

1) работоспособность предложенной методики не нуждается в доказательстве, поскольку она является логическим следствием способа организации процесса синтеза алгоритма – в виде цепочки стандартных операторов обработки изображения;

2) исследование эффективности предложенной методики является предметом самостоятельного дополнительного исследования; изначально же очевидно, что методика позволяет автоматизировать процесс синтеза лучшего для текущих условий съемки алгоритма предварительной обработки изображений, и это весьма положительный эффект.

### Заключение

Основная значимость статьи состоит в том, что в ней изложена новая методика полностью автоматического синтеза цепочек алгоритмов, улучшающих качество градиентных изображений. В основе разработанной методики лежат следующие новые и оригинальные идеи.

1. Использование формализма – *системы аксиом* Цермело-Френкеля на начальном этапе формирования групп функциональных операторов (алгоритмов), из которых формируются цепочки; в состав каждой функциональной группы включается пустой оператор; указанные новации позволяют осуществлять корректное формирование функциональных групп в автоматическом режиме.

2. Автоматическая *одновременная* структурная и параметрическая идентификация синтезируемой цепочки операторов.

3. Применение в качестве механизма реализации автоматической идентификации синтезируемой цепочки операторов *модифицированного* генетического алгоритма. Модификация состоит в совмещении процедуры поиска с одновременным ведением динамического множества Парето и обеспечивает возможность «промышленного» (без участия эксперта) использования предлагаемой методики.

4. Применение *двух* критериев качества синтезируемой цепочки операторов:

1) точность восстановления контура изделия;

2) минимум трудоемкости реализуемых операций (времени обработки изображения цепочкой).

5. Применение *новой* методики двухкритериального оптимального выбора решения для задач с динамической генерацией вариантов, т.е. с заранее неопределённым числом вариантов. Методика создана для реализации двухкритери-

альности процесса синтеза цепочки операторов. Она является *самостоятельно значимой*, – может использоваться в различных приложениях двухкритериального оптимального выбора.

Теоретическими результатами статьи являются:

1) методика полностью автоматического синтеза цепочек алгоритмов, улучшающих качество градиентных изображений;

2) модифицированный генетический алгоритм, совмещающий процедуру поиска с одно-временным ведением динамического множества Парето;

3) методика двухкритериального оптимального выбора решения для задач с динамической генерацией вариантов;

её концептуальные элементы:

– критерий максимума относительного преимущества;

– теоремы о множестве Парето с двумя показателями качества элементов;

– понятия рационального максимума и минимума двух качественно разнонаправленных критериев;

– ограничения на критерии;

– её процедурные элементы;

– генерация вариантов;

– ведение динамического множества Парето;

– принятие решения по критерию максимума относительного преимущества.

Созданная методика автоматического синтеза цепочек алгоритмов и контекстно разработанная самостоятельно значимая методика двухкритериального оптимального выбора решения имеют и теоретическое, и практическое значение, поскольку пополняют состав методического обеспечения практики решения задач, связанных как с обработкой изображений, так и принятием оптимальных решений.

#### Библиографический список

1. Гонсалес Р., Вудс Р. Цифровая обработка изображений. Издание 3-е, исправленное и дополненное М.: Техносфера. 2012. 1104 с.

2. Munteanu C., Rosa A. Gray-Scale Image Enhancement as an Automatic Process Driven by Evolution. IEEE Transactions on Systems, Man, and Cybernetics. 2004. V. 34. P. 1292-1298.

3. Махно Т. А. Автоматическое построение комбинаций алгоритмов обработки изображений в задаче обработки УЗ изображений сонных артерий. ГрафиКон' 2014: 24-я Международная конференция по компьютерной графике и зрению: Ростов-на-Дону, Академия архитектуры и искусств ЮФУ Труды конференции. 2014. С. 104-107.

4. Махно Т. А. Автоматизированная система обработки ультразвуковых изображений сонных артерий на основе эволюционных алгоритмов: Дис. ... канд. техн. наук. Славянск. 2016. 152 с. // Авторефераты диссертаций: [сайт]. URL: <http://abstracts.donnu.edu.ua/article/view/2465> (дата обращения: 27.11.2018).

5. Белоусов А. А., Спицын В. Г., Сидоров Д. В. Применение генетических алгоритмов и вейвлет-преобразований для повышения качества изображений. Известия Томского политехнического университета. 2006. Выпуск № 7. Т. 309. С.21-26.

6. Чернявский А. В., Спицын В. Г. Применение эволюционирующей нейронной сети для улучшения качества изображений. Известия Томского политехнического университета. – 2006. Т. 309. № 7. с. 26-31.

7. Белоусов А. А., Спицын В. Г. Двухэтапный метод улучшения изображений // VI Всероссийская научно-практическая конференция «Технологии Microsoft в теории и практике программирования». Секция № 4 «Интеллектуальные системы и технологии». Томск, 17 – 18 марта 2009 г. 2009. С. 123-125.

8. Терёхин А. В. Система распознавания трехмерных объектов.: Дис. ... канд.техн. наук. защищена: 30.09.2015, утверждена: 30.10.2015, Муром, 2014. 177 с.

9. Кузнецов А. Е., Рыжиков А. С. Регрессионный алгоритм формирования непрерывного изображения по данным кадровой съемки. Вестник Рязанского государственного радиотехнического университета. 2014. № 48. С. 11-14.

10. Белов В. В., Лопатин А. К. Алгоритм оценки геометрических параметров изделий, находящихся на конвейерной ленте. XXI век: итоги прошлого и проблемы настоящего плюс: Периодическое научное издание. Пенза: Изд-во Пенз. гос. технол. акад. 2016. Выпуск № 06(34) С. 34-42.

11. Петрушин В. Н., Ульянов М. В. Планирование экспериментального исследования трудоёмкости алгоритмов на основе бета-распределения. Информационные технологии и вычислительные системы. 2008. № 2. С. 81-91.

12. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. М.: Вильямс. 2016. 1328 с. 263 ил.

UDC 004.932.4:004.421

## APPLICATION OF GENETIC ALGORITHMS FOR THE FORMATION OF ALGORITHM CHAINS FOR IMAGE IMPROVING

**V. V. Belov**, Dr. Sc. (Tech.), full professor, RSREU, Ryazan; vvbeloff@yandex.ru

**A. K. Lopatin**, post-graduate student, SUHSS, Kolomna; ak\_lopatin@mail.ru

*The task of forming gradient images of products located on a conveyor belt in the process of implementing automatic verification of their geometric dimensions is considered. **The aim of the work** is to develop a method for automatic synthesis of an algorithm for obtaining high-quality gradient images. The algorithm is synthesized as the chain of necessary procedures (operators) used in practice of solving image processing problems. Structural and parametric identification of the synthesized chain is carried out automatically using a modified genetic algorithm. Input parameters of the synthesized chain are: the image of inspected product in RGB format and binary image-standard. The synthesized chain adapts to shooting conditions (camera type, its resolution, surface illumination), the image formed by it is intended for the subsequent selection of product edges in order to evaluate its geometric parameters. In the process of chain synthesis, the vector of its operators and their parameters is formed*

**Keywords:** genetic algorithms, image processing, algorithms compositions, set of algorithms, image filtering, edge detecting.

**DOI:** 10.21667/1995-4565-2018-66-4-1-78-89

### References

1. **Gonsales R., Vuds R.** Цифровая обработка изображений. Издание 3-е, исправленное и дополненное, (Digital image processing) Moscow, Tehnosfera. 2012. 1104 p. (in Russian).
2. **Munteanu C., Rosa A.** Gray-Scale Image Enhancement as an Automatic Process Driven by Evolution. IEEE Transactions on Systems, Man, and Cybernetics. 2004, vol. 34. pp. 1292-1298.
3. **Mahno T. A.** Avtomaticheskoe postroenie kombinacij algoritmov obrabotki izobrazhenij v zadache obrabotki UZ izobrazhenij sonnyh arterij. GrafiKon' 2014: 24-ja Mezhdunarodnaja konferencija po komp'juternoj grafike i zreniju. Rostov-na-Donu, Akademija arhitektury i iskusstv JuFU Trudy konferencii. 2014, pp. 104-107. (in Russian).
4. **Mahno T. A.** Avtomatizirovannaja sistema obrabotki ul'trazvukovyh izobrazhenij sonnyh arterij na osnove jevoljucionnyh algoritmov (Automated system for processing ultrasound images of carotid arteries on the basis of evolutionary algorithms.). Dis. ... kandtehn. nauk. Slavjansk. 2016. 152 p. (in Russian).
5. **Belousov A. A., Spicyn V. G., Sidorov D. V.** Primenenie geneticheskikh algoritmov i vejvlet-preobrazovanij dlja povyshenija kachestva izobrazhenij. Tomsk, Izvestija Tomskogo politehnicheskogo universiteta. 2006, no. 7. vol. 309, pp. 21-26. (in Russian).
6. **Chernjavskij A. V., Spicyn V. G.** Primenenie jevoljucionirujushhej nejronnoj seti dlja uluchshenija kachestva izobrazhenij. Tomsk, Izvestija Tomskogo politehnicheskogo universiteta. – 2006, vol. 309, no. 7, pp. 26-31. (in Russian).
7. **Belousov A. A., Spicyn V. G.** Dvuhjetapnyj metod uluchshenija izobrazhenij. VI Vserossijskaja nauchno-prakticheskaja konferencija «Tehnologii Microsoft v teorii i praktike programirovaniya». Sekcija no. 4 «Intel'lektual'nye sistemy i tehnologii». Tomsk, 17-18 marta 2009 g. 2009, pp. 123-125. (in Russian).
8. **Terjohin A. V.** Sistema raspoznavanija treh-mernyh ob'ektov (The system of recognition of three-dimensional objects):. Dis. ... kandtehn. nauk. Murom., 2014. 177 p. (in Russian).
9. **Kuznecov A. E., Ryzhikov A. S.** Regressionnyj algo-ritm formirovaniya nepreryvnogo izobrazheniya po dannym kadrovoj s"emki. Vestnik Ryazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2014, no. 48, pp. 11-14.
10. **Belov V. V., Lopatin A. K.** Algoritm ocenki geometricheskikh parametrov izdelij, nahodjashihhsja na konvejernoj lente. XXI vek: itogi proshlogo i problemy nastojashhego plus: Periodicheskoe nauchnoe izdanie. Penza, Izd-vo Penz. gos. tehnol. akad. 2016, no. 06(34), pp. 34-42. (in Russian).
11. **Petrushin V. N., Ul'janov M. V.** Planirovanie jeksperimental'nogo issledovanija trudomkostj algoritmov na osnove beta-raspredelenija. Informacionnye tehnologii i vychislitel'nye sistemy. FRC CSC RAS. 2008, no. 2, pp. 81-91. (in Russian).
12. **Kormen T., Lejzerson Ch., Rivest R.** Algoritmy: postroenie i analiz. Moscow, Williams. 2016. 1328 p. 263 il. (in Russian).