

ПЕРЕКЛЮЧАЮЩИЙСЯ МЕДИАННЫЙ ФИЛЬТР С БЛОКОМ ПРЕДВАРИТЕЛЬНОГО ДЕТЕКТИРОВАНИЯ

Приоров А.Л., Апальков И.В., Бухтояров С.С., Хрящев В.В.

Введение

На практике часто встречаются изображения, искаженные импульсным шумом. Причинами возникновения таких помех на изображении могут быть сбои в работе канального декодера, связанные с замиранием сигналов в канале связи или перемещением абонентов, шум видеодатчика, зернистость пленки и т.д.

При воздействии импульсного шума на изображении с оттенками серого цвета наблюдаются белые или (и) черные точки, хаотически разбросанные по кадру [1-3]. Хорошие результаты для сохранения перепадов оттенков, различных границ и локальных пиков яркости на искаженных импульсным шумом изображениях может дать применение медианных фильтров (МФ) [4].

Анализ источников по вопросам медианной фильтрации импульсного шума [5,6] показывает, что основными недостатками данного метода обработки являются:

- ослабление сигнала, что проявляется на изображении в виде размытых контуров деталей;
- повреждение неискаженных («хороших») пикселей изображения.

Для устранения рассмотренных недостатков предложен ряд модификаций МФ [7-12]. На сегодняшний день наибольший интерес у исследователей вызывает переключающаяся схема [13-15]. Идея данной модификации медианного фильтра основана на дополнительном шаге обнаружения импульсов (детектировании импульсов). После того как позиции импульсов обнаружены, для вычисления медиан или других локальных характеристик используются только «хорошие» пиксели. На этапе обнаружения импульсов может быть применена нечеткая логика [16], самоорганизующиеся нейронные сети [17] и другие методы [18].

Целью работы является усовершенствование алгоритмов удаления импульсного шума. Предложенный алгоритм способен эффективно удалять такой шум даже из сильно зашумленных изображений, обладая при этом относительно низкой вычислительной сложностью.

Для оценки качества восстановления использовались среднеквадратическая ошибка (СКО) и пиковое отношение сигнал-шум (ПОСШ), определяемые выражениями [19]:

$$\text{СКО} = \frac{1}{N} \sum_i (u_i - \varphi_i)^2, \quad \text{ПОСШ} = 20 \log_{10} \frac{255}{\sqrt{\text{СКО}}},$$

где N - общее число всех пикселей изображения, u_i и φ_i -

Предложен переключающийся медианный фильтр с блоком предварительного детектирования для восстановления изображений, искаженных импульсным шумом типа «соль-и-перец». Проведен сравнительный анализ предложенного алгоритма фильтрации с пятью известными модификациями медианных фильтров как с точки зрения среднеквадратической ошибки восстановления, так и с точки зрения визуальных оценок. Приведены примеры обработки тестовых изображений указанными цифровыми фильтрами.

значения пикселей в позиции i на восстановленном и исходном изображениях соответственно. Схема восстановления изображения приведена на рис. 1.

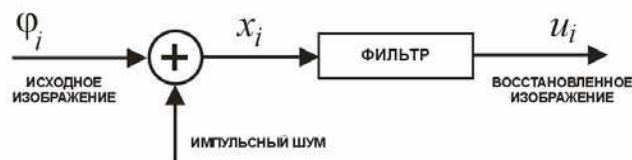


Рис. 1. Схема восстановления изображения

Для удобства рассмотрим сначала общую схему алгоритма фильтрации, а затем более детально остановимся на работе его основных блоков.

Общая схема алгоритма

Рассматриваемый шум представляет собой импульсный биполярный шум вида соль-и-перец (salt-and-pepper), описываемый следующей математической моделью:

$$x_i = \begin{cases} 0 & \text{с вероятностью } p_n; \\ 255 & \text{с вероятностью } p_p; \\ \varphi_i & \text{с вероятностью } 1 - (p_n + p_p), \end{cases}$$

где $p_n = p_p = 0.5R$, R - коэффициент зашумленности ($0\% \leq R \leq 100\%$), φ_i отображает значения «хороших» пикселей, 0 - фиксированное значение отрицательных выбросов, 255 - фиксированное значение положительных выбросов и x_i отображает значения поврежденных пикселей изображения.

Предлагаемый медианный фильтр с нейросетевым детектором (НПМ-фильтр) использует переключающуюся схему. При этом импульсный детектор состоит из двух блоков: первый блок - включает процедуру предварительного обнаружения импульсов и второй - процедуру нейросетевой коррекции предварительных результатов. Процедура предварительного детектирования позволяет найти большинство импульсов для выбранной модели шума, но имеет серьезный

недостаток - "хорошие" пиксели, значения которых совпадают со значениями "соли" (255) или "перца" (0) автоматически определяются как импульсы. Процедура нейросетевой коррекции позволяет различать такие пиксели и используется для корректировки результата предварительного детектирования. Далее проводится процедура фильтрации с учетом информации, полученной на этапе нейросетевой коррекции.

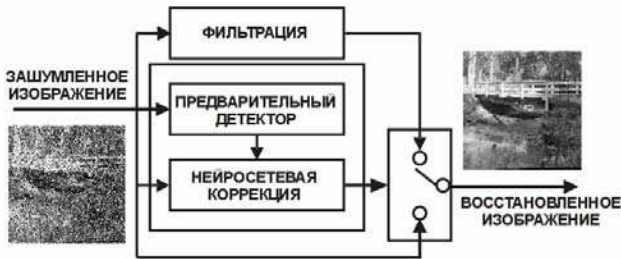


Рис. 2. Общая схема НПМ алгоритма

Общая схема предлагаемого алгоритма приведена на рис. 2. Рассмотрим этапы его работы более подробно.

ПРЕДВАРИТЕЛЬНЫЙ ДЕТЕКТОР

Алгоритм предварительного обнаружения импульсов использует два изображения. Первое представляет собой поврежденное полутоновое изображение $\{x_i\}$, которое отображает значение пикселя в позиции $i = (i_1, i_2)$. Второе - бинарное изображение $\{f_i\}$, где значение f_i показывает, является ли пиксель в позиции i импульсом или нет, т.е. $f_i = 0$ означает, что пиксель i «хороший», а $f_i = 1$ означает, что пиксель i - импульс. В начале полагаем, что все пиксели изображения «хорошие», т.е. $f_i \equiv 0$.

Далее для каждого пикселя x_i находим минимальное и максимальное значения в пределах окна $W_D \times W_D$ (W_D - нечетное целое, не меньше трех). Пусть Ω_i^W отображает $W \times W$ окно с центром вокруг i

$$\Omega_i^W = \{j = (j_1, j_2) | i_1 - (W - 1)/2 \leq j_1 \leq i_1 + (W - 1)/2, i_2 - (W - 1)/2 \leq j_2 \leq i_2 + (W - 1)/2\},$$

тогда

$$\min_i = \min\{x_j | j \in \Omega_i^{W_D}\},$$

$$\max_i = \max\{x_j | j \in \Omega_i^{W_D}\}.$$

Далее используем простое соотношение для определения, является ли текущий пиксель импульсом или нет

$$f_i = \begin{cases} 0, & \text{если } \min_i < x_i < \max_i; \\ 1, & \text{иначе.} \end{cases}$$

Полученное бинарное изображение $\{f_i\}$ является результатом процедуры предварительного обнаружения импульсов.

НЕЙРОСЕТЕВАЯ КОРРЕКЦИЯ

Процедура нейросетевой коррекции использует три изображения. Первое отображает поврежденное полутоновое изображение $\{y_i\}$. Второе представляет бинар-

ное изображение $\{f_i\}$, полученное на этапе предварительного обнаружения импульсов. И третье, также бинарное изображение $\{g_i\}$, используется для записи финального результата процедуры нейросетевой коррекции. В начале полагаем, что $\{g_i\}$ равно предварительному результату детектирования, т.е. $g_i \equiv f_i$.

Для каждого пикселя, определенного предварительной процедурой как импульс, применяется нейронная сеть. Во время обучения нейронной сети на тренировочных данных обнаружено, что большинство информации о том, является ли пиксель импульсом или нет, содержится в семи локальных характеристиках пикселя. Это значение самого пикселя, отклонения от медиан и дисперсии для различных окрестностей рассматриваемого пикселя. Таким образом, размерность входного вектора равна семи. Для вычисления этих величин используем только «хорошие» пиксели, т.е. пиксели с $f_i = 0$. Пусть M^W отмечает число пикселей с $f_i = 0$ в пределах окна $W \times W$. Если M^W четное, то медиана вычисляется как среднее арифметическое двух средних элементов отсортированных данных. Тогда определяем элементы входного вектора нейронной сети v следующим образом:

$$\begin{aligned} v_0 &= y_i; \\ v_1 &= \text{Med}\{y_j | f_j = 0, j \in \Omega_i^3\} - y_i; \\ v_2 &= \text{Disp}\{y_j | f_j = 0, j \in \Omega_i^3\}; \\ v_3 &= \text{Med}\{y_j | f_j = 0, j \in \Omega_i^5\} - y_i; \\ v_4 &= \text{Disp}\{y_j | f_j = 0, j \in \Omega_i^5\}; \\ v_5 &= \text{Med}\{y_j | f_j = 0, j \in \Omega_i^7\} - y_i; \\ v_6 &= \text{Disp}\{y_j | f_j = 0, j \in \Omega_i^7\}, \end{aligned}$$

где Med и Disp обозначают вычисление медианы и дисперсии в заданных окрестностях соответственно.

Выход нейронной сети представляет собой одно значение, отображающее два возможных состояния для пикселя (пиксель «хороший» или «плохой»). В алгоритме используется трехслойный персептрон с S_D нейронами в скрытом слое. Усредненные зависимости СКО восстановленного изображения от параметра нейронной сети S_D для набора тестовых изображений приведены на рис. 3.

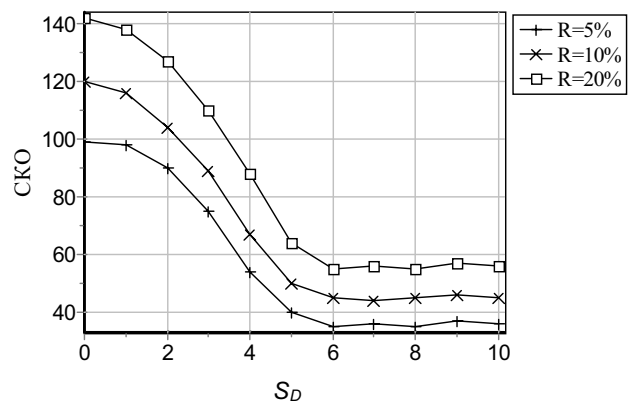


Рис. 3. Зависимость СКО от числа нейронов в скрытом слое

Видно, что параметр S_D не чувствителен к изменению степени зашумленности изображения, и лучшие результаты получаются, начиная от значения $S_D=6$.

Пусть D_i отображает выходное значение нейронной сети в диапазоне $[0; 1]$ для пикселя в позиции i , причем, если D_i близко к 1, то пиксель был определен импульсом правильно, а если к 0, то пиксель с большой вероятностью «хороший». Тогда используем простое соотношение для записи выходного скорректированного значения:

$$g_i = \begin{cases} 0, & \text{если } f_i = 1; D_i < 0.1; M^3 > 0; \\ g_i, & \text{иначе.} \end{cases}$$

Полученное бинарное изображение $\{g_i\}$ является результатом работы процедуры нейросетевой коррекции. Общее влияние нейронной сети на выполнение алгоритма для серии тестовых изображений приведено на рис. 4.

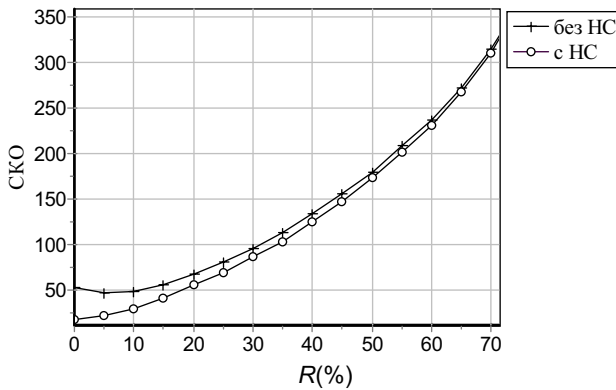


Рис. 4. Зависимость СКО восстановленного изображения от плотности импульсного шума при модификации детекторной части алгоритма

Приведенная зависимость показывает, что включение нейронной сети дает значительный положительный эффект, особенно при небольших значениях шума, что проявляется в лучшем сохранении границ объектов изображения.

ПРОЦЕДУРА ФИЛЬТРАЦИИ

В ходе процедуры фильтрации генерируются две последовательности изображений. Первая представляет собой последовательность изображений $\{\{z_i^{(0)}\}, \{z_i^{(1)}\}, \dots, \{z_i^{(n)}\}, \dots\}$, где $z_i^{(0)}$ - входное зашумленное изображение, а $z_i^{(n)}$ отображает значение пикселя в позиции i после n -ой итерации. Вторая - это последовательность бинарных изображений $\{\{h_i^{(0)}\}, \{h_i^{(1)}\}, \dots, \{h_i^{(n)}\}, \dots\}$, где бинарное значение $h_i^{(n)} = 0$ означает, что пиксель i «хороший», а $h_i^{(n)} = 1$, что он «плохой». Начальное изображение $\{h_i^{(0)}\}$ равно результату процедуры нейросетевой коррекции $\{g_i\}$, т.е. $h_i^{(0)} \equiv g_i$.

На n -ой итерации ($n=1, 2, \dots$) для каждого пикселя $z_i^{(n-1)}$ сначала вычисляются медианные значения $m_i^{(n-1)}$ в окне $W_F \times W_F$ (W_F - нечетное, не меньше трех) с центром вокруг пикселя. Медианы вычисляются, используя только «хорошие» пиксели (с $h_i^{(n-1)} = 0$) в пределах окна. Пусть M отмечает число всех пиксе-

лей с $h_i^{(n-1)} = 0$ в окне $W_F \times W_F$. Если M четное, то медиана вычисляется как среднее арифметическое между двумя средними элементами отсортированных данных. Если $M > 0$, то $m_i^{(n-1)} = \text{Med}\{z_j^{(n-1)} | h_j^{(n-1)} = 0, j \in \Omega_i^{W_F}\}$.

Значение $z_i^{(n)}$ изменяется только, если пиксель i - импульс, следующим образом

$$z_i^{(n)} = \begin{cases} m_i^{(n-1)}, & \text{если } h_i^{(n-1)} = 1; M > 0; \\ z_i^{(n-1)}, & \text{иначе.} \end{cases}$$

Если импульсный пиксель был изменен, то дальше он рассматривается как «хороший», т.е.

$$h_i^{(n)} = \begin{cases} h_i^{(n-1)}, & \text{если } z_i^{(n)} = z_i^{(n-1)}; \\ 0, & \text{если } z_i^{(n)} = m_i^{(n-1)}. \end{cases}$$

Процедура останавливается на N_F -ой итерации, когда все пиксели модифицированы, т.е.

$$\sum_i h_i^{(N_F)} = 0.$$

Полученное в результате изображение $\{z_i^{(N_F)}\}$ и есть восстановленное изображение.

РЕЗУЛЬТАТЫ МОДЕЛИРОВАНИЯ

На рис. 5 приведены результаты восстановления тестового изображения «Ручей и мост» («Stream and Bridge») [20] шестью различными алгоритмами: 1) медианным фильтром (Мед) с окном 3×3 ; 2) адаптивным медианным (АМ) фильтром; 3) прогрессивным переключающимся (ППМ) фильтром; 4) итеративным медианным фильтром (ИМ) с окном 3×3 и числом итераций, равным 10; 5) центрально взвешенным медианным фильтром (ЦВМ) с окном 5×5 и центральным весом, равным 3; 6) предложенным нейросетевым переключающимся (НПМ) фильтром.

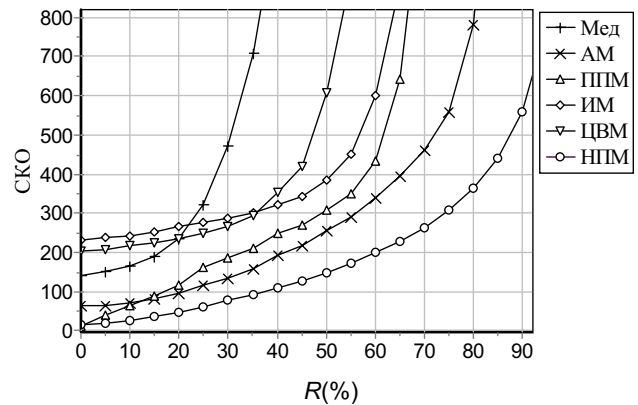


Рис. 5. Результаты восстановления тестового изображения «Ручей и мост» для шума типа «соль-и-перец»

Таким образом, для модели импульсного шума типа «соль-и-перец» предложенный НПМ алгоритм демонстрирует лучшие результаты на всем диапазоне степеней зашумления.

На рис. 6 показаны результаты восстановления различными алгоритмами тестового изображения «Лена» [20], зашумленного 30% шумом указанного типа.



Рис. 6. Результаты восстановления тестового изображения различными алгоритмами: (а) исходное изображение «Лена»; (б) изображение, искаженное 30% шумом типа «соль и перец»; (в) медианный фильтр с окном 3×3 ; (г) ЦВМ фильтр с окном 5×5 и центральным весом, равным 3; (д) ИМ фильтр с окном 3×3 и числом итераций равным 10; (е) ППМ фильтр; (ж) АМ фильтр; (з) предложенный НПМ фильтр

Стандартный медианный фильтр (рис. 6в) хорошо сохраняет детали изображений, но при этом на нем остается много шумовых пикселей. ЦВМ фильтр (рис. 6г) работает лучше медианного, но также пропускает много импульсов. ИМ фильтр (рис. 6д) удаляет больше импульсов, чем ЦВМ фильтр, но изменяет много хороших пикселей, что выражается в смазывании изображения. ППМ фильтр (рис. 6е) не способен удалить блоки импульсов (скопления шумовых пикселей). АМ фильтр (рис. 6ж) демонстрирует высокие результаты. Он удаляет большинство импульсов, сохраняя детали, но повреждает границы объектов. Наилучший результат получен с помощью НПМ алгоритма (рис. 6з). При заданном уровне зашумления (30%) он удаляет все поврежден-

ные пиксели, хорошо сохраняя детали и границы объектов изображения.

Демонстрация эффективного восстановления двух сильно зашумленных тестовых изображений «Элейн» («Elein») и «Танк» («Tank») [20] приведена на рис. 7 и рис. 8 соответственно.

На рис. 7 приведен пример восстановления тестового изображения «Элейн», поврежденного 50% импульсным шумом типа «соль-и-перец». Сравниваются результаты обработки трех разновидностей медианного фильтра (стандартный медианный фильтр с маской 5×5 , АМ и НПМ фильтры). Видно, что наилучшие результаты восстановления достигнуты при использовании предложенного НПМ фильтра.



(а) ПОСШ=8,46 дБ



(б) ПОСШ=24,63 дБ

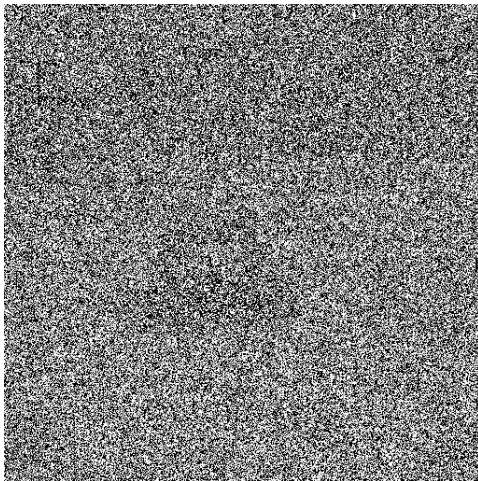


(v) ПОСШ=30,7 дБ

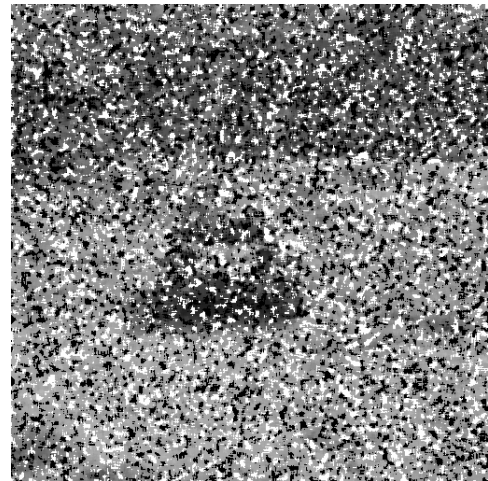


(r) ПОСШ=32,98 дБ

Рис. 7. Результаты восстановления сильно зашумленного изображения «Элейн»:
(a) поврежденное 50% импульсным шумом; (б) восстановленное медианным фильтром с маской 5x5;
(в) восстановленное АМ фильтром; (г) восстановленное НПМ фильтром



(a) ПОСШ=6,67 дБ



(б) ПОСШ=10,74 дБ



(в) ПОСШ=19,89 дБ



(г) ПОСШ=27,41 дБ

Рис. 8. Результаты восстановления сильно зашумленного изображения «Танк»:
(a) поврежденное 80% импульсным шумом; (б) восстановленное медианным фильтром с маской 5x5; (в) восстановленное АМ фильтром; (г) восстановленное НПМ фильтром

На рис. 8 тестовое изображение «Танк», искаженное уже 80% импульсным шумом типа «соль-и-перец», также восстанавливалось указанным набором медианных фильтров. Отметим, что если на исходном изображении рис. 7 основные контуры объектов еще угадываются, то на исходном изображении рис. 8 они уже почти незаметны. Видно, что при столь сильном зашумлении стандартный медианный фильтр с поставленной задачей практически не справился. Тем не менее, предложенный НПМ фильтр успешно справился с задачей восстановления, убрав шум и лишь относительно несильно исказив контуры объектов.

Заключение

Разработанная новая модификация медианного фильтра обладает высокими характеристиками и может применяться в различных устройствах цифровой обработки изображений, работающих в сложной сигнально-помеховой обстановке. Описанный алгоритм в большинстве случаев полностью убирает импульсный шум типа «соль и перец» и хорошо сохраняет границы объектов изображения. По этой причине он может быть использован на этапе предобработки в устройствах распознавания и анализа изображений, где требования к качеству изображений часто бывают достаточно высокими.

Предложенный алгоритм медианной фильтрации по предварительным расчетам допускает эффективную аппаратную реализацию на современных сигнальных процессорах и микроконтроллерах, в частности, на программируемых сигнальных микроконтроллерах серии «МУЛЬТИКОР» [21-22].

Литература

1. Красильников Н.Н. Цифровая обработка изображений. – М.: Вузовская книга, 2001.
2. Mitra S., Sicuranza G. Nonlinear Image Processing. Academic Press, 2001.
3. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2005.
4. Хуанг Т.С. Быстрые алгоритмы в цифровой обработке изображений: преобразования и медианные фильтры. – М.: Радио и Связь, 1984.
5. Pitas I., Venetsanopoulos A. Nonlinear Digital Filters: Principles and Applications. - Boston, MA: Kluwer, 1990.
6. Мушкаев С.В. Реализация ранжирующих и медианных фильтров на процессоре NM6403 // Цифровая обработка сигналов. 2004. № 4. С. 44-46.
7. Хрящев В.В., Соколенко Е.А., Звонарев П.С., Куйкин Д.В. Усовершенствование алгоритмов восстановления изображений на основе ранговой статистики // Докл. 7-й Международ. конф. «Цифровая обработка сигналов и ее применение» (DSPA-2005). Москва, 2005. Т. 2, С. 304-306.
8. Ko S., Lee Y. Center weighted median filters and their applications to image enhancement // IEEE Trans. Circuits systems. 1991. V. 38, № 9. P. 984-993.
9. H. Hwang, R. Haddad. Adaptive median filters: new algorithms and results // IEEE Trans. on image processing. 1995. V. 4, № 4. P. 499-502.
10. L. Yin, R. Yang, M. Gabbouj, Y. Neuvo. Weighted median filters: a tutorial // IEEE Trans. Circuits systems. 1996. V. 43, № 3. P. 157-192.
11. T. Nodes, N. Gallagher. Median filters: some modifications and their properties // IEEE Trans. ASSP. 1982. V. 30, № 5. P. 739-746.
12. E. Abreu, M. Lightstone, S. Mitra, K. Arakawa. A new efficient approach for the removal of impulse noise from highly corrupted images // IEEE Trans. on image processing. 1996. V. 5, № 6. P. 1012-1025.
13. Wang Z., Zhang D. Progressive switching median filter for the removal of impulse noise from highly corrupted images // IEEE Trans. Circuits systems – II. 1999. V. 46, № 1. P. 78-80.
14. Apalakov I., Khryashchev V., Priorov A., Zvonarev P. Image denoising using adaptive switching median filter // Proc. IEEE int. conf. on image processing (ICIP'05). Genoa. Italy. 2005. V. I, P. 117-120.
15. Apalakov I., Khryashchev V., Priorov A., Zvonarev P. Adaptive switching median filter with neural network impulse detection step // Proc. of the 15th international conference on artificial neural networks (ICANN-2005). Warsaw. Poland. 2005. LNCS 3696, Springer-Verlag, P. 537-542.
16. Zhang D., Wang Z. Impulse noise detection and removal using fuzzy techniques // Electron. Lett. 1997. V. 33, P. 378-379.
17. Kong H., Guan L. A neural network adaptive filter for the removal of impulse noise in digital images // Neural networks. 1996. V. 9, № 3. P. 373-378.
18. R. Chan, C. Ho, M. Nikolova. Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization // IEEE Transaction on Image Processing. 2005. V. 14, № 10. P. 1479-1485.
19. Сэлмон Д. Сжатие данных, изображений и звука. – М.: Техносфера, 2004.
20. The University of Southern California. Signal and Image Processing Institute Image Database (<http://sipi.usc.edu/database>).
21. Петричкович Я., Солохина Т. Цифровые сигнальные контроллеры «МУЛЬТИКОР» - новые отечественные серии систем на кристалле // Докл. 6-й Международ. конф. «Цифровая обработка сигналов и ее применение» (DSPA-2004). Москва. 2004. Т. 1, С. 8-15.
22. Солохина Т., Александров Ю., Петричкович Я. Сигнальные контроллеры компании «ЭЛВИС»: первая линейка отечественных DSP // Электроника: Наука, Технология, Бизнес. 2005. № 7. С. 70-77.