

Восстановление изображения на дефектных пикселях КМОП и ПЗС матриц

С. С. Магазов

МГТУ им. Баумана

Аннотация. В статье исследуется задача восстановления растрового изображения на дефектных областях ПЗС или КМОП матриц. Задача восстановления изображения делится на подзадачи: восстановление контуров и восстановление текстур. Эти задачи решаются специальной машиной восстановления изображений, которая использует адаптированные к задаче методы классификации изображений, нейронную сеть и методы восстановления изображения. Предложен оригинальный метод восстановления изображений в видеоряде. Проведен анализ вычислительных сложностей использованных методов.

Ключевые слова: генерация текстур, аппроксимация дискретных функции, статистики изображения, параметры Харалика и Лавса, фильтр Габор, нейронные сети.

DOI 10.14357/20718632190303

Введение

Одна из основных проблем производства ПЗС или КМОП матриц состоит в том, что современные технологии дают малый процент выхода матриц, не имеющих дефектные пиксели. Матрицы с небольшим количеством дефектных пикселей (несколько сотен) можно использовать при условии, что в обработчике изображения предусмотрены методы, восстанавливающие изображение. Очевидно, что правдоподобная программная коррекция изображения дефектных пикселей может дать большую материальную выгоду, что и обуславливает важность проблемы восстановления изображений.

Задача восстановления изображения на дефектных пикселях ПЗС или КМОП матрицы формулируется следующим образом: дано изображение с дефектами, то есть областями, на которых изображение отсутствует. Требуется правдоподобно и согласованно со смежными фрагментами изображения восстановить изобра-

жение на этих областях. Эта задача чем - то напоминает задачу генерации текстур и интерполяции поверхностей. Разработано множество разнообразных методов решения такого рода задач [1], которые дают хорошие результаты для определенных типов изображений, но далеко не все подходят для задачи восстановления. Так, самообучающиеся конкурентные нейронные сети GAN, к которым в настоящее время привлечено внимание специалистов, не позволяют решать задачу восстановления. Классическая модель GAN состоит из *генератора изображения G* , *дискриминатора D* и *потока истинных изображений S* . Изображения строятся путем внесения случайных искажений в реальные изображения, в частности, используя заранее заданный класс вариаций. Если попытаться применить GAN к задаче восстановления, то возникает проблема с организацией потока S . Подробные материалы по архитектуре и алгоритму функционирования GAN можно найти в [2].

В статье будем рассматривать восстановление изображения как следующие две взаимосвязанные задачи: восстановление контуров и восстановление текстуры. Эта задача решается специальной *машиной восстановления изображений*. Машина имеет набор экстракторов классификационных признаков, нейронную сеть, решающую задачу выбора наиболее подходящего метода восстановления изображения, и подсистему, реализующую задачу восстановления.

В качестве классификационных признаков используется ряд специально подобранных параметров Харалика [3, 5], Лавса [6] и фрактальные размерности [7]. Для более тонкого исследования используется фильтр Габора [11-12]. Все классификационные признаки составляют *поле признаков*, которое используется при решении задачи классификации.

В задачах классификации часто используется пороговый подход, который предполагает сложную процедуру определения значений порогов для признаков. К недостаткам этого подхода следует отнести сложность построения механизма адаптации к особенностям изображения. В статье предлагается классифицировать изображения с помощью предварительно обученной на поле признаков нейронной сети. После чего машина восстановления выполняет один из приведённых ниже методов восстановления изображения.

Статистический синтез изображений предполагает выбор статистической модели с оптимальным набором параметров [13-15]. Параметры вычисляются на основе изучения статистических зависимостей в изображении. После того как модель построена и осуществлена настройка параметров, необходимо провести статистическое моделирование дефектной области изображения.

В статье рассматриваются простые модели восстановления, построенные на основе гистограммы изображения и модели, использующие GLCM матрицы Харалика. Результат восстановления будет удачным, если изображения обладают необходимым набором свойств, куда входит статистическая однородность, отсутствие сложных фигур и т.д. Эти свойства характерны для изображений искусственного происхождения.

Для сложных естественных изображений метод не дает хороших результатов.

Тайлинговое моделирование [16] - это непараметрический метод, основная идея которого заключается в построении из фрагментов недефектной части изображения, изображение дефектной области. Сложность применения тайлинга заключается в разработке метода автоматического выделения недефектных фрагментов и синтеза алгоритма их комбинирования. Метод дает хорошие результаты для искусственно заданных текстур,

Video Row модель восстановления изображения. В статье предлагается новый непараметрический метод восстановления изображения с помощью анализа видеоряда, полученного во время непрерывной съемки камерой. Восстановление изображения в видеоряде учитывает закономерности изменения изображения. Идея предложенного метода состоит том, чтобы для восстановления изображения на текущем кадре использовать фрагменты кадров предыдущих кадров. Этот дает хорошие результаты для всех типов изображения, но накладывает требования на расположение дефектной области и скорости обновления изображения.

1. Необходимые определения

В качестве математической модели ПЗС или КМОП матрицы будем использовать *растровую карту* (растр). Растр представляет собой прямоугольную матрицу *пикселей* $M(n, m)$. Пиксель может принимать целочисленные значения, которые назовем *градаций серого*. Кроме этого, пиксель может быть *дефектным*, то есть не иметь целочисленного значения (null). Пиксель будем обозначать парой его $\langle i, j \rangle$ координат. Наряду с пикселем будем рассматривать его непосредственную *окрестность* (Рис. 1)

	135°	90°	45°	
5	6	7		
4	$\langle i, j \rangle$	0	0°	
3	2	1		

Рис. 1. Нумерация пикселей окрестности и направлений (по Фриману)

Определение 1.1.

1. Пиксель $\langle k, l \rangle$ назовем *смежными* для пикселя $\langle i, j \rangle$, если выполняется одно из следующих условий:

1. $k=i+1$ смежные по грани (направление 0°),
2. $k=i-1$ смежные по грани (направление 180°),
3. $l=j+1$ смежные по грани (направление 90°),
4. $l=j-1$ смежные по грани (направление 270°),
5. $k=i+1, l=j+1$ - смежные по правому верхнему углу (направление 45°),
6. $k=i-1, l=j-1$ - смежные по левому нижнему углу (направление 225°),
7. $k=i+1, l=j-1$ - смежные по правому нижнему углу (направление 315°),
8. $k=i-1, l=j+1$ - смежные по левому верхнему углу (направление 135°).

Заметим, что пиксели, расположенные на границе растра, имеют менее восьми соседей.

Определение 1.2.

1. Произвольное множество пикселей $E \subseteq M$ называется *областью*.

2. Множество пикселей L назовем *тонким контуром*, если каждый пиксель смежен не более чем с двумя пикселями из L . Число пикселей, принадлежащих контуру $n=|L|$, назовем его *длиной*.

3. *Тонкой прямой* назовем контур, пиксели которого располагаются по горизонтали, либо по вертикали, либо по диагонали.

4. Связанное множество пикселей k назовем *компонентой*, если любые два пикселя k связываются кривой из точек k , в частности, тонкий контур является *компонентой*.

5. *Пятно* - это компонента, которая не пиксель и не кривая.

6. *Окном* назовем прямоугольную область растра. Ориентация сторон может быть любой, относительно его сторон.

7. Число пикселей, принадлежащих области $n=|E|$, назовем *площадью области*.

8. *Расстояние* между пикселями d_{ij} и d'_{kl} будем вычислять по формуле

$$\rho(d_{ij}, d'_{kl}) = \max\{|i - k|, |j - l|\}$$

9. *Окрестностью радиуса ε* пикселя $d=\langle i, j \rangle$ назовем множество:

$$\delta(d, \varepsilon) = \{d' \mid \rho(d, d') \leq \varepsilon\}$$

Например, пиксели смежные с d образуют его окрестность радиуса один.

10. Пиксели окрестности, лежащие на прямой, проведенной через центр окрестности и лежащие по разные стороны центра, назовем *противоположными*.

11. Пусть F – подмножество растра и Σ – некоторое множество подмножеств растра. Семейство подмножеств $\sigma \subseteq \Sigma$ - назовём *покрытием* F , если $F \subseteq \bigcup_{x \in \sigma} x$.

Все приведенные выше определения носят конструктивный характер и могут быть алгоритмически проверены.

Дадим определения, которые будут использоваться для анализа изображения.

Определение 1.3.

1. Целочисленную частичную функцию $J(i, j)$, определенную на растре назовем *изображением* $J(i, j)$.

2. Область, на которой $J(i, j)$ не определена, назовем *дефектной областью*.

3. Назовем $l=|F| \setminus |M|$ *степенью дефектности* $J(i, j)$, где $|F|$ – мощность дефектной области, $|M|$ – мощность растра.

4. Функцию $J(i, j)$ назовем *ε, r -непрерывной* на пикселе, если $|J(i, j) - J(i', j')| < \varepsilon$, где $\langle i', j' \rangle \in \delta(\langle i, j \rangle, r)$.

5. Функцию $J(i, j)$ назовем *ε, r -непрерывной по направлению θ* , если

$$|J(i, j) - J(i', j')| < \varepsilon, \text{ где } \langle i', j' \rangle \in \delta(\langle i, j \rangle, r) \text{ по направлению } \theta.$$

6. *Односторонней ε, r границей* назовем тонкий контур l , вдоль которого функция изображения ε -разрывная с одной стороны, то есть $|J(i, j) - J(i', j')| > \varepsilon$, где $i, j \in l$, i', j' лежат по одну сторону кривой и принадлежат окрестности радиуса менее r .

7. *Двухсторонней ε, r границей* назовем контур l , вдоль которой функция изображения ε -разрывная, то есть $|J(i, j) - J(i', j')| > \varepsilon$ и $|J(i, j) - J(i'', j'')| > \varepsilon$, где $i, j \in l$ и точки i', j', i'', j'' лежат по разные стороны l и принадлежат окрестности радиуса менее r .

Приведённые определения границы носят частный характер, так, не только резкое изменение яркости, но и изменение типа текстуры дают зрительное восприятие границы.

Проанализируем области изображения, которые распознаются глазом как контур. Рассмотрим график функции изображения $J(x, y)$



Рис. 2. Изображение дома с дефектной областью

(Рис. 2), определенной на прямоугольном фрагменте R , содержащем фрагмент границы.

На Рис. 3 отчетливо виден скачок яркости на границе между текстурами неба и стены. Шумы и искажения в матрице и других элементах камеры делают изображения контура размытым. Размытость носит случайный характер, поэтому для описания контура l будем использовать статистические модели. «Реальная» контур на изображении имеет ширину более одного пикселя и интенсивность серого меняется от точки к точке. Будем рассматривать степень интенсивности серого и ширину линии как случайные величины, распределённые согласно нормальным законам распределения. Расположение контура на растре будет представляться тонким контуром, который можно получить хорошо известной в теории распознавания образов операцией утончения. Назовём его *направляющей контур* и представлять его кодировкой Фримана $\gamma(k)$ [17], где k - номер точки контура.

Дадим определение статистической модели контура, которой будем придерживаться в этой работе.

Определение 1.4.

Статистическая модель контура – это пятерка: $l = \langle N(h, \sigma_h), N(w, \sigma_w), \gamma, \langle i, j \rangle, h * \exp(-\alpha d) \rangle$, где h – средняя интенсивность серого,

w – средняя ширина контура,

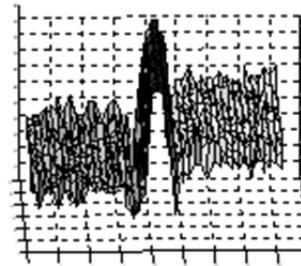
σ_h – среднее квадратическое отклонение интенсивности серого,

σ_w – среднее квадратическое отклонение ширины контура,

i, j – координаты пикселя, к которому привязывается γ ,

α коэффициент затухания;

$N(m, \sigma)$ – нормальное распределение.

Рис. 3. Функция изображения $J(i,j)$ определенная на R (см. рис. 2)

d – расстояние точки контура от направляющего контура

В предложенной модели формула $h * \exp(-\alpha d)$ позволяет учитывать затухания яркости. Предложенная модель контура отличается от широко известной модели зашумлённого контура [3], что объясняется требованием нашей задачи. Нам необходимо моделировать продолжения контура.

Ниже будем использовать следующие широко используемые определения обозначения:

1. $\hat{J}(i, j)$ – преобразование Фурье $J(i, j)$.
2. $E(J(i, j))$ – среднее значение (математическое ожидание).
3. $D(J(i, j))$ – дисперсия $J(i, j)$.
4. $\sigma(J(i, j))$ – среднее квадратическое отклонение $J(i, j)$.

2. Постановка задачи восстановления

Уточним постановку задачи восстановления изображений. Начнем с перечисления гипотез, в рамках которых решается задача восстановления изображения:

- отсутствует априорная информация о восстанавливаемом изображении;
- матрица может работать в режиме видеооряда, то есть генерировать упорядоченный по времени набор изображений, в котором смежные изображения мало отличаются друг от друга;
- степень дефектности менее десяти процентов;
- дефектные компоненты расположены нерегулярно;
- шумы и искажения матрицы и других элементов камеры делают изображение размытым;

– восстановление изображения должно происходить в реальном масштабе времени (несколько десятков миллисекунд).

Пусть на растре M задано изображение $J(i, j)$ и $F \subseteq M$ – область дефектных пикселей, на которых $J(i, j)$ не определено. Разобьём F на *дефектные компоненты* $k_1 \dots k_m$ (связанные области). Дефектные компоненты могут сильно отличаться по форме, мы будем использовать следующую грубую классификацию по форме: *изолированный пиксель, кривая, пятно*. Для упрощения изложения каждое пятно впишем в аппроксимирующий прямоугольник со сторонами, параллельными сторонам растра. Везде ниже будем считать, что каждый k_i – прямоугольник.

Для того, чтобы восстановить изображение, будем анализировать изображения в некотором *расширенном окне* $k_i \subseteq W(k_i)$. Положим $W(k_i)$ подобным k_i в пропорции 1:2, что, на наш взгляд, достаточно для восстановления k с высокой степенью достоверности. Будем использовать $W^-(k_i) = W(k_i) \setminus k_i$ для обозначения множества недефектных точек из $W(k_i)$.

Так, поставленная задача восстановления, как и задача аппроксимации или генерации изображений, имеет множество возможных решений, критерии выбора будут уточняться по ходу изложения.

3. Классификационные признаки текстур

Здесь рассмотрим ряд экстракторов классификационных признаков изображения. Каждый из признаков в отдельности имеет невысокую разделяющую способность (70%-80%).

3.1. Экстракторы признаков регулярности текстур

Для установления наличия регулярностей в $J(i, j)$ будем использовать корреляционные признаки Харалика и корреляционные зависимости, вычисленные непосредственно для $J(i, j)$:

Функция авторегрессии изображения

Авторегрессия вычисляется для окна $W(k)$. Определим скользящее квадратное окно w со стороной δ . Окно будем двигать слева направо и сверху вниз внутри $W(k)$ и вычислять в нем функцию авторегрессии следующим образом:

$$K(x, y, k, l) = \frac{\sum_{i=k-\delta}^{k+\delta} \sum_{j=l-\delta}^{l+\delta} (J(i, j) J(i-x, j-y))}{\sum_{i=k-\delta}^{k+\delta} \sum_{j=l-\delta}^{l+\delta} (J(i, j))^2}$$

Наличие ярко выраженных пиков у $K(x, y, k, l)$ говорит в пользу наличия регулярностей в текстуре. Этот метод требует подбора оптимальной величины окна. Мы будем увеличивать размеры окна (δ) пока авторегрессия увеличивается, выберем в качестве оптимального размера окна δ на котором функция авторегрессии начинает уменьшаться.

Если использовать преобразования Фурье для вычисления числителя дроби, то сложность вычисления $K(x, y, k, l)$ будет порядка $O(n \log(n))$.

Корреляционные коэффициенты GLCM матрицы

Для оценки регулярности текстуры будем использовать корреляционные коэффициенты, вычисленные для матрицы GLCM Харалика. Пусть $p(i, j)$ – GLCM матрица для расстояния d и направления α . \square вычисленная для окна $W(k)$. Корреляционные коэффициенты вычисляются по следующим формулам:

1. Коэффициенты корреляции:

$$Hr_3 = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (ij) p(i, j) - \mu_i \mu_j}{\sigma_i \sigma_j}$$

где $\mu_i = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i) p(i, j)$

$$\mu_j = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (j) p(i, j)$$

$$\sigma_i = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - \mu_i)^2 p(i, j)$$

$$\sigma_j = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (j - \mu_j)^2 p(i, j)$$

Высокие значения Hr_7 и Hr_6 . Аргументом в пользу наличия регулярности.

Энтропии

Энтропия оценивает степень структурной однородности изображения и вычисляется по формулам:

$$Hr_7 = - \sum_{i=1}^n p_{x+y}(i) \log(p_{x+y}(i))$$

$$Hr_{11} = - \sum_{i=1}^n p_{x-y}(i) \log(p_{x-y}(i)),$$

где

$$p_{x+y}(k) = \sum_{i=1}^{N_g} \sum_{\substack{j=1 \\ |i+j|=k}}^{N_g} p(i, j)$$

$$и \ p_{x-y}(k) = \sum_{i=1}^{N_g} \sum_{\substack{j=1 \\ |i-j|=k}}^{N_g} p(i, j)$$

Высокое значение энтропий будет аргументом в пользу стохастичной природы текстуры.

Спектральная характеристика

Один из методов классификации текстуры основан на анализе спектра Фурье изображения $J(i, j)$. Спектр вычисляется по следующей формуле:

$$\hat{J}(n, m) = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} J(k, l) \exp(i2\pi(\frac{kn}{N_1} + \frac{lm}{N_2}))$$

Наличие ярко выраженных максимумов у $|\hat{J}(n, m)|$ позволяет говорить о наличии регулярности в текстуре. Сложность вычисления быстрого преобразования Фурье $O(MN^2+NM^2)$, что слишком затратно, для дефектных областей больших размеров.

3.2. Экстрактор энергетических характеристик Лавса

Метод Лавса [19] позволяет извлекать локальную информацию об изображении. Метод заключается в свертке изображения с центрально симметрическими ядрами:

1. L5= [1 4 6 4 1] - сглаживает изображение (Local averaging);
2. E5=[-1 -2 0 2 1]-выделяет края, (Edge);
3. S5=[-1 0 2 0 -1]-обнаруживает пятна (Spot);
4. W5=[-1 2 0 -2 1] - для обнаружения волн (Wave);
5. R5=[1 4 6 -4 1] - обнаруживает периодические составляющие изображения (Ripple detection).

Эти ядра перемножаются, в результате чего получаются новые ядра (Табл. 1.)

Полученные ядра сворачиваются с расширенным окном $W(k)*X5_Y5$. По совокупности девяти матриц оцениваются энергетическая

Табл. 1. Ядра энергетических характеристик

L5_L5= L5L5	E5_L5= E5L5	S5_L5= S5L5
L5_E5= L5E5	E5_E5= E5E5	S5_E5= S5E5
L5_S5= L5S5	E5_S5= E5S5	S5_S5= S5S5

характеристика пикселя. Для нашей задачи будем использовать только ядра E5_S5, S5_S5 и E5_E5.

Быстрые алгоритмы свертки, как известно, имеют сложность порядка $O(n \log(n))$, где $n=|k|$.

3.3. Экстрактор на основе фильтра Габора

2D-фильтр Габора используется для обнаружения на изображении ориентированных структурных элементов. Рассмотрим фильтр Габора на расширенном окне $W(k)$ и пусть размер окна $M \times N$. Фильтр Габора для изображения $J(i, j)$ вычисляется по следующей формуле:

$$FG(f_0, \theta, \gamma, \mu, \psi, \gamma) = \sum_{k=1}^M \sum_{l=1}^N J(k, l) G(k, l; f_0, \theta, \gamma, \mu, \psi)$$

Ядро G вычисляется по следующей формуле:

$$G(k, l; f_0, \theta, \psi, \gamma, \mu) = \exp(\frac{f_0^2}{\gamma^2} x'^2 + \frac{f_0^2}{\mu^2} y'^2) \exp(i2\pi x' + \psi),$$

где $x' = k \cos(\theta) + l \sin(\theta)$ и $y' = -k \sin(\theta) + l \cos(\theta)$

- θ – ориентация фильтра ядра;
- f_0 – центральная частота фильтра;
- k, l – масштаб шкал по осям x, y ;
- γ, μ – степень остроты Гауссиана по осям;
- ψ – смещение фазы ядра.

Если в $J(i, j)$ присутствуют структурные элементы, ориентированные по направлению θ , то фильтр Габора, ориентированный в направлении θ , дает сильный отклик. В нашем случае считаем, что θ может принимать значение $0^0, 45^0, 90^0, 135^0$ градусов. Методом перебора находим максимальное и минимальное значение θ .

Параметры f_0, γ, μ, ψ в функционале $FG()$ оцениваем следующим образом:

- $f_0 = kw$, где w - частота с максимальной амплитудой преобразования Фурье $\widehat{J(k, l)}$, а параметра k подбирается экспериментально,
- $\gamma = k * \max_j ((J(i, j) - E(J(i, j)))^4 / \sigma(J(i, j))^4)$ – максимальный эксцесс по оси j , k подбирается экспериментально,
- $\mu = k * \max_x ((J(i, j) - E(J(i, j)))^4 / \sigma(J(i, j))^4)$ – максимальный эксцесс по оси i , k подбирается экспериментально,
- фаза, которая подбирается экспериментально.

3.4. Экстрактор фрактальных характеристик

Для анализа текстур со сложной геометрией воспользуемся методами фрактальной геометрии. Под изображением со сложной геометрией часто понимают объединение неперекрывающихся множеств самоподобных копий фигур, возможно с небольшими статистическими вариациями. Мы здесь будем понимать фрактальное изображение как сложноустроенное множество состоящих из множества изолированных точек и линий.

3.4.1. Размерности Минковского (Minkowski dimension)

Будем рассматривать $J(i, j)$ как многообразие в трехмерном евклидовом пространстве R^3 . Пусть Σ_δ - множество кубов со стороной δ . Пусть $\sigma_\delta \subseteq \Sigma_\delta$ покрытие $J(i, j)$ такие, что центр кубов принадлежит текстуре и σ_δ имеет минимальную мощность текстуры. Введем обозначение $N_\delta(J(x, y)) = |\sigma_\delta|$.

Определение 3.4.1.1. Верхняя и нижняя размерность $J(x, y)$ (BCD-Box-Computing Dimension) определяется следующим образом:

$$\overline{dim}_b(J(i, j)) = \overline{\lim}_{\delta \rightarrow 0} \frac{\log(N_\delta(J(i, j)))}{\log(1/\delta)} \quad \text{верхняя BCD- размерность } J(x, y)$$

$$\underline{dim}_b(J(i, j)) = \underline{\lim}_{\delta \rightarrow 0} \frac{\log(N_\delta(J(i, j)))}{\log(1/\delta)} \quad \text{нижняя BCD- размерность } J(x, y)$$

Заметим, что в этом определении не используется понятие меры. Введем альтернативное определение размерности, которое более удобно для вычислений и которое использует меру. Для этого нам потребуется понятие параллельного тела (Parallel Bodies).

Определение 3.4.1.2 δ - параллельное тело поверхности $J(i, j)$ это:

$$P_\delta = \{x \in R^3: |x-y| < \delta \text{ для некоторого } y \in J(i, j)\}.$$

Как видно из определения, δ -параллельное тело покрывает поверхности $J(i, j)$. Определим s -размерность $J(i, j)$. Пусть $Vol(P_\delta)$ -объем параллельного тела.

$$\overline{dim}_m(J(i, j)) = 3 - \overline{\lim}_{\delta \rightarrow 0} \frac{\log(Vol(P_\delta))}{\log(\delta^{3-s})}$$

верхняя s - размерность $J(i, j)$

$$\underline{dim}_m(J(i, j)) = 3 - \underline{\lim}_{\delta \rightarrow 0} \frac{\log(Vol(P_\delta))}{\log(\delta^{3-s})}$$

нижняя s - размерность $J(i, j)$

Следующая теорема говорит об условии эквивалентны dim_b и dim_m .

Теорема 3.4.1.1. Если $\overline{dim}_m(J(x, y)) = \underline{dim}_m(J(x, y))$ $J(i, j)$ не пустое и ограниченное множество в R^3 , тогда $dim_b(J(x, y)) = dim_m(J(x, y))$.

Польза этой теоремы состоит в том, что dim_m размерность считать проще, чем dim_b . В [18] предложен метод MFS (Modified FractalSignature) вычисления dim_m .

Определим две поверхности $u_\delta(i, j)$ и $b_\delta(i, j)$, которые аппроксимируют $J(i, j)$ снизу и сверху и находятся на расстоянии δ от нее следующим образом:

$$u_0(i, j) = b_0(i, j) = J(i, j)$$

$$u_\delta(i, j) = \max \left\{ u_{\delta-1}(i, j) + 1, \max_{|(m, n) - (i, j)| \leq 1} u_{\delta-1}(m, n) \right\}$$

$$b_\delta(i, j) = \max \left\{ b_{\delta-1}(i, j) - 1, \max_{|(m, n) - (i, j)| \leq 1} b_{\delta-1}(m, n) \right\}$$

Вычислим объем параллельного тела P_δ , как $Vol(P_\delta) = \sum_{i, j} (u_\delta(i, j) - b_\delta(i, j))$

Пусть $\{\delta_i\}_{i < \infty}$ - некоторая последовательность, стремящаяся к нулю. В качестве классификационного признака будем брать монотонный участок из n членов последовательности $\left\{ \frac{\log(Vol(P_{\delta_i}))}{\log(\delta_i^{3-s})} \right\}$ как приближение к размерности dim_m .

Нетрудно заметить, что вычисление $Vol(P_\delta)$ имеет сложность $O(n)$, n -количество точек в области определения $J(i, j)$.

3.5.2. s-размерность Хаусдорфа (Hausdorff Measure)

Для полноты картины дадим определение размерности Хаусдорфа, которая имеет большее теоретическое значение.

Пусть $J(i, j)$ и $s, \delta > 0$. Определим функцию: $H_\delta^s(J(i, j)) = \inf \{ \sum_{i=1}^{\infty} diam(x_i)^s : 0 < |diam(x_i)| < \delta, \text{ для всех } x_i \}$,

где $\{x_i\}$ - покрытие многообразия $J(i, j)$

$diam(x_i)$ - диаметр x_i

Определим s -размерность Хаусдорфа: $H^s(J(x, y)) = \lim_{\delta \rightarrow 0} H_\delta^s(J(x, y))$.

В формуле для H_δ^s \inf берется по всем покрытиям радиуса δ многообразия $J(i, j)$, то есть требуется провести перебор все покрытий радиуса

менее δ . Сложность такого перебора превосходит экспоненциальную сложность поэтому s -размерность Хаусдорфа мало интересна для практики.

4. Классификация изображений

Так как нет априорной информации о свойствах изображения, то восстановлению будет предшествовать анализ W с целью определения наиболее эффективного метода восстановления. Анализ состоит в классификации изображения на W . Классы изображений представлены иерархией на Рис. 4.

Дадим неформальное разъяснение относительно принципов формирования классификации изображений, которые мы будем использовать в нейронной сети (п. 6.). Иерархия строится на основе экстрадированных признаков. Каждый из признаков в отдельности обладает различительной способностью, которая варьирует в пределах 60%-80% относительно нашего класса иерархий. Такая надёжность классификации недостаточна для наших целей. Поэтому будем использовать *поле признаков*, то есть совокупности всех признаков, полученных от экстракторов.

Каждый из нами рассматриваемых признаков является числом или числовой функцией. Традиционно определение класса дается через пороги. Здесь предлагается вместо этого обучать нейронную сеть (многослойный перцептрон с тремя слоями) на основании специально подобранной БД изображений. Для этого экспертами согласно предложенной иерархии

классов формируется БД. Обученная сеть и даст нам классификатор изображений.

5. Модели восстановления изображения

Рассмотрим восстановление изображения как следующие взаимосвязанные задачи: восстановление линий и восстановление текстур.

5.1. Восстановления линий дефектных пикселей

Рассмотрим задачу восстановления линии без самопересечений, состоящей из дефектных пикселей (Рис. 5).

Предлагается следующий алгоритм. Для каждого пикселя d дефектной линии вычисляем СКО окрестности $\delta(d, 2)$ относительно точки $i, j \in \gamma$.

Далее анализируется среднеквадратическое отклонение.

1. Если значение СКО небольшое, тогда для нахождения $J(i, j)$ применим либо метод осреднения по окрестности, либо вычислим $J(i, j)$ с помощью медианного фильтра.

$$J(i, j) = (1/8) \sum_{\substack{\delta(d,1) \exists \langle k,l \rangle \\ \text{не дефектные} \\ \text{пиксели}}} J(k, l)$$

2. Если СКО велико, то анализируем противоположные пары пикселей из окрестности. Выбираем пару с наименее отличающимися значениями.

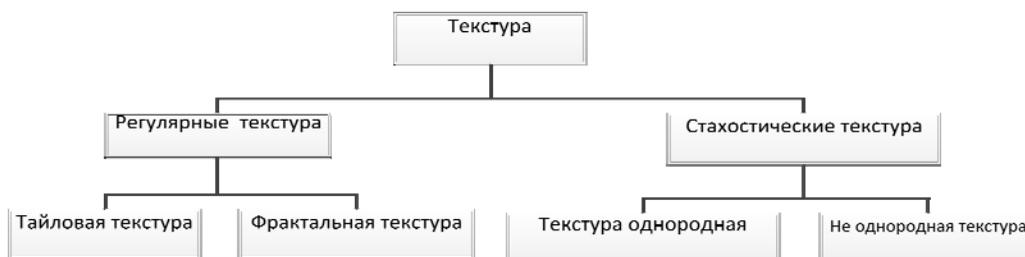


Рис. 4. Иерархии классов

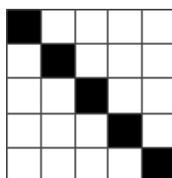


Рис. 5. Пример линии дефектных точек

ниями и усредняем значение по этой паре. Полученное значение выбираем в качестве $J(i,j)$.

Изолированный дефектный пиксель рассматривается как частный случай линии.

5.2. Модель восстановления непрерывности контура

Пусть в расширенном окне W содержатся два контура l_1 и l_2 , которые прерваны дефектной компонентой k . Возможно, что l_1 и l_2 являются отрезками одного контура, в этом случае назовем их *сопоставимыми* или они заканчиваются в k , тогда назовем их *несопоставимыми*. Требуется правдоподобно соединить сопоставимые l_1 и l_2 , контуром l на дефектных пикселях k или закончить эти контуры в k . Наша версия восстановления линии будет состоять из следующих этапов:

- распознавание направляющих γ_1 и γ_2 для контуров l_1 и l_2 ;
- определение сопоставимости l_1 и l_2 ;
- восстановление непрерывности направляющих;
- вычисление статистических моментов l ;
- построение контура $l_1 \cup l_2$.

Распознавание направляющих γ_1 и γ_2 для контуров l_1 и l_2 . В W^- направляющие γ_1 и γ_2 выделяются одним из хорошо известных и достаточно надёжных методов. Например, в качестве таких методов можно использовать конволюцию $W^- * G$, где G ядро (Собеля, Робертса, Превитта и т.д.) с последующим бинаризацией изображения. В результате должны получить направляющие γ_1 и γ_2 , где $n=1 \dots M1$ и $\gamma_2(m)$, где $m=1 \dots M2$, представленные в форме Фримана и точки привязки i_1, j_1 и i_2, j_2 контуров.

Определение сопоставимости l_1 и l_2 . Пусть d_0 и d_1 точки касания γ_1 и γ_2 к компоненте k . Зададим интервал $0 < \alpha < b$ в котором лежит угол α .

В общих чертах алгоритм сопоставления выглядит следующим образом. Строим в d_i угол α такой, что касательная к контуру γ_i в d_i будет биссектриса угла α (Рис. 6). Далее возможны следующие случаи: Внутри α отсутствующую точку $d_{(i+1) \bmod 2}$, тогда увеличиваем угол α до тех пор, пока не появится $d_{(i+1) \bmod 2}$. или $\alpha = b$. Если точка $d_{(i+1) \bmod 2}$ появилась в угле α (Рис. 6), то контура *сопоставимы* иначе *несопоставимы*. Угол α назовем *углом рассогласования*.

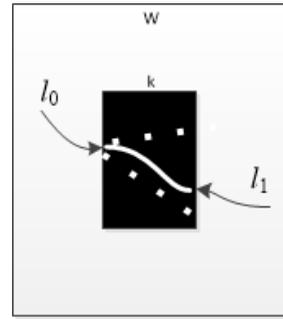


Рис. 6. Восстановление непрерывности линии

Восстановление несопоставимых направляющих. Пусть точки касания d_1 и d_2 конечными точками l_1 и l_2 .

Восстановление сопоставимых контуров. Используя любой подходящий метод интерполяции, соединяем точки касания d_1 и d_2 контуром γ . И строим общую направляющую, как $\gamma' = \gamma_1 \cup \gamma \cup \gamma_2$.

Определение статистических моментов контура l . Вычислим моменты для $l_i, i=1,2$. Среднее значение градации серого для l_i вычисляем следующим образом: $h^i = E(J(\gamma_i(k)))$.

Среднеквадратическое отклонение интенсивности серого вдоль направляющей γ_i оценим, как

$$\sigma_h^i = \sqrt{\frac{\sum (J(\gamma_i(j)) - h^i)^2}{M_i}}$$

Средняя ширина w_i определяется следующим образом: задается ε и берется точка на $\gamma_i(k)_i$ и находится наименьший радиус r_k , при котором точка является односторонней ε, r -границей $J(\gamma_i(k))$. Это радиус и будем считать шириной контура в точке.

$$w^i = \frac{\sum_{k=1}^{Mq} w_k^i}{|\gamma_i|}$$

где w_k^i – ширина в точке контура γ_i .

Среднеквадратическое отклонение по ширине контура l_i вычислим как

$$\sigma_w^i = \frac{\sum_{j=1}^{M_i} \sigma(\gamma_i(j))}{|\gamma_i|}$$

где $\sigma(\gamma_i(j))$ вычисляем в окрестности точки $\gamma_i(j)$ для радиуса ε вдоль направляющей.

Вычисляем статистические параметры контура l следующим образом:

$$h = (h^1 M_1 + M_2 h^2) \frac{1}{M_1 + M_2}$$

$$\text{и } \sigma_h = (\sigma_h^1 M_1 + \sigma_h^2 M_2) \frac{1}{M_1 + M_2}$$

$$w = (w^1 M_1 + M_2 w^2) \frac{1}{M_1 + M_2}$$

$$\text{и } \sigma_w = (\sigma_w^1 M_1 + \sigma_w^2 M_2) \frac{1}{M_1 + M_2}$$

Веса пропорциональны длине контуров.

В результате получим модели контура $l = \langle N(h, \sigma_h), N(w, \sigma_w), \gamma, \langle i, j \rangle \rangle$. Коэффициент затухания определяется экспертным путем.

Построение контура $l_1 \cup l_2$. Методом статистического моделирования восстанавливаем l , затем проводим сглаживание на стыках l с l_1 и l_2 .

5.4. Модели восстановления текстур

Здесь предлагается методы восстановления текстур. Каждый из предложенных методов хорошо работает для определённого типа текстур.

5.4.1. Статистические модели восстановления

Рассмотрим параметрические и непараметрические статистические модели текстур. Обсудим их вычислительные сложности.

Восстановление текстур на основе гистограммы

Восстановление стохастических текстур на основе гистограммы - самый простой способ восстановления текста. Нормализованная гистограмма GLH (Graj Level Histogram) строится для окна W . Диапазон серого разбивается на интервалы. GLH представляет частотность появления интервалов яркости. $J(i, j)$ По GLH вычисляются различные статистики. Приведём ряд часто используемых статистик:

1. Выборочное среднее значение яркости $M = \sum_{i=1}^n \text{GLH}(h_i) h_i$;
2. Выборочные моменты n -го порядка: $\sum_{i=1}^n (h_i - M)^n \text{GLH}(h_i)$.

Далее GLH рассматривается как статистический аналог одного из заранее заданных распределений $f_1(h, a_1 \dots a_l) \dots f_n(h, a_1 \dots a_l)$, где h - значение яркости и $a_1 \dots a_l$ - параметры распределения. Выборочные моменты и другие параметры распределения можно использовать как для классификации текста. Набор параметров зависит от типа распределения. После чего вычисляются критерии согласия χ^2 , на основании которых выбираются наиболее подходящие распределения. Полученное таким образом

распределение будем считать статистической моделью текстуры.

Методом статистического моделирования, используя построенную модель, восстанавливается текстура. Этот метод дает хорошие результаты для стохастических текстур. Если текстура регулярная, то достоверно ее восстановить предложенным методом не представляется возможным.

Сложность построения GLH имеет порядок $O(n)$, сложность вычисления статистик и имитационного моделирования так же имеет порядок $O(n)$.

Восстановление текстур с помощью GLCM матрицы

Харалик предложим распознавать текстуры, основываясь на GLCM (Graj level co-occurrence matrices) матрице [12, 13]. Мы предлагаем использовать GLCM матрицы для восстановления текстуры.

Пусть $P_{\theta d}(i, j)$ - нормированная GLCM матрица размерности $N_g \times N_g$, построенная для окна W^- , расстояния $d=1$ и угла $\theta \in \{0, 45^\circ, 90^\circ, 135^\circ\}$. Будем рассматривать $P_{\theta d}$, как двумерную дискретную плотность распределения, где случайной величиной являются пары значений яркости. Используя $P_{\theta d}$, определим условное распределение вероятности яркостей:

$$P_{\theta d}(i|j) = \frac{P_{\theta d}(i, j)}{\sum_{i=0}^{N_g} P_{\theta d}(i, j)}$$

Алгоритм восстановления значений яркости дефектных пикселей использует методы статистического моделирования. Будем считать, что вычислены GLCM матрицы P_θ для $\theta = 0, 45^\circ, 90^\circ, 135^\circ$ и $d=1$.

1. Пусть d_{kl} недефектный пиксель, у которого есть смежный дефектный пиксель, который находится под углом θ и на расстоянии 1;
2. Выбираем этот дефектный пиксель и строим распределение $p(i|J(k, l))$;
3. Моделируем случайную величину согласно распределению $P_{\theta d}(i|J(k, l))$
4. Значение, полученное в результате моделирования, будет значением выбранной дефектной точки, считаем ее недефектной.
5. Процесс повторяется для всех дефектных точек.

Значение одного и того же пикселя может быть смоделировано с использованием распре-

делений, полученных для разных направлений. Например, рассмотрим следующее расположение пикселей, на нижнем ряду - недефектные пиксели, на верхнем ряду - дефектные пиксели (Рис. 7) Тогда для определения значения для $d_{k+1,i}$ можно использовать распределение $P_{45,1}(i|J(k, m - 1))$, $P_{90,1}(i|J(k, m))$ и $P_{135,1}(i|J(k, m + 1))$, полученные для GLCM матриц по соответствующим направлениям.

То есть результат работы алгоритма зависит от выбора начального пикселя и угла, под которым он виден из недефектного пикселя. В этом случае необходимо доопределить порядок вычисления значений яркости дефектных пикселей.

Вычислительная сложность построения GLCM достаточно велика и равна $64n^2 + O(n)$, где n - количество анализируемых пикселей.

5.4.2. Синтаксические модели восстановления

В общем случае *тайл* представляет собой повторяющийся фрагмент текстуры, как правило, фрагмент имеет форму правильного многоугольника. Тайловые текстуры построены из ограниченного набора *тайлов*. Пусть удалось выделить тайлы из W^- , и $T = \langle t_1 \dots t_n \rangle$ имена этих тайлов. Будем рассматривать процесс восстановления текстур как процесс копирования и сшивания тайлов. Для этого на множестве тайлов определим частичные бинарные операции склейки $\langle o_1 \dots o_m \rangle$. Например,

- $o_1(t_1, t_2)$ - операция подклеивания сверху тайла t_2 к тайлу t_1 ,
- $o_2(t_1, t_2)$ операция подклеивания снизу тайла t_2 к тайлу t_1 , и т.д...

Процесс восстановления изображения будет управляться словом, которое выводится в специально построенной грамматике $G = (N, \Sigma, P, S)$, которую определим следующим образом:

- $N = \{A_1 \dots A_k, S\}$ - конечное множество *нетерминальных* символов, которые будем обозначать заглавными буквами;
- $\Sigma = \{t_1 \dots t_n, o_1 \dots o_m\}$ - конечное множество *терминальных* символов (операции и тайлы);
- S - начальный нетерминальный символ;
- P - следующие правила вывода: $S \rightarrow t_1 | \dots | t_n$;
 $S \rightarrow SAS | (S)$; $A \rightarrow o_1 | \dots | o_m$

Все необходимые определения по грамматикам можно найти в [1]. Очевидно, что язык

$?(k+1, m-1)$	$?(k+1, m)$	$?(k+1, m+1)$
$J(k, m-1)$	$J(k, m)$	$J(k, m+1)$

Рис. 7. Пример расположения дефектных и недефектных пикселей

грамматики G состоит из правильно построенных выражений, состоящих из бинарных операций, имен тейлов и скобок. Пусть $f(t_1 \dots t_n)$ некоторое такое выражение. При помощи этого $f(t_1 \dots t_n)$ построим текстуру, выполняя последовательно слева направо операции из $f(t_1 \dots t_n)$. Если операция невыполнима, что процесс останавливается. В результате получим в определенном смысле регулярную текстуру. Процесс начинаем с границы дефектной компоненты. Предложенный подход хорошо работает, если края тейлов сочетаются. Поэтому после операций подклеивания предполагается производить гауссовское сглаживание шва.

Операция склейки и фильтрации не требует больших вычислительных ресурсов. Сложность использования метода заключается в определении набора тайлов.

Общая схема восстановления текстуры на ленте состоит из следующих этапов:

1. Выделение тайлов из W^- .
2. Вывод множества слов L в G , обеспечивающий заполнение дефектной компоненты.
3. Восстановление дефектной компоненты согласно L .
4. Сглаживание швов текстуры фильтром гаусса.

5.5. Video Row модель восстановления

Video Row модель предназначена для восстановления кадров видеоряда. Будем считать, что видеоряд получен во время непрерывной съемки камерой. Видеоряд имеет ряд особенностей, которые позволяют с высоким качеством восстанавливать произвольные изображения.

Идею метода объясним с помощью следующего примера. Пусть камера равномерно движется справа налево с известной скоростью v и ведет непрерывную съемку с периодом одна

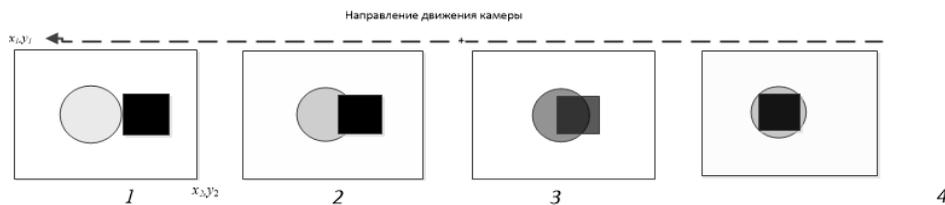


Рис. 8. Последовательность кадров. Индексы изображения показывают моменты времени

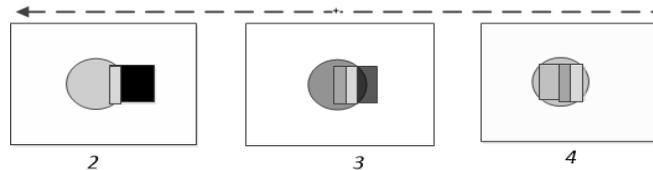


Рис. 9. Процесс восстановления дефектной компоненты

секунда. На Рис. 8 представлена последовательность кадров $J_1(i,j), J_2(i,j), J_3(i,j), J_4(i,j)$, которую формирует камера в моменты времени $k=1...4$. Дефектный компонент изображен черным квадратом $R(x_1,y_1,x_2,y_2)$, где x_1,y_1 - координата левого верхнего угла, x_2,y_2 - координата правого нижнего угла. Круг изображает объект сцены, который неподвижен и в момент начала движения ($t=1$) и координаты центра x_r,y_r . Круг меняет цвет в некоторые моменты времени t_1,t_2,t_3,t_4 внутри секундных интервалов.

Ставится задача - используя кадры, восстановить изображение на дефектной области.

Рассмотрим, как изменяется положение объекта. В момент времени 2 изображение сместится на $d=v \cdot I$ вправо, соответственно, координата центра объекта относительно раstra поменяется и станет равной x_r-l,y_r . Для восстановления изображения необходимо выделить прямоугольный фрагмент $R_l(x_1-d,y_1,x_2,y_2)$ в момент времени 1 и наложить его на дефектную компоненту, как это показано на Рис. 9. В момент времени t_3 сдвигаем установленный на предыдущем шаге прямоугольник на d вправо затем на растре, который получен в момент времени t_2 выделяем $R_l(x_1-d,y_1,x_2,y_2)$, и накладываем его на место сдвинутого прямоугольника, далее процесс повторяется до конца видеоряда.

Итак, на основании примера не трудно понять, что для восстановления изображения необходимо знать скорость перемещения изображения на растре. Вопрос состоит в получении скорости перемещения изображения. Скорость можно получать от внешних датчиков или путем математических расчетов, например, путем минимизации следующего функционала:

$$\Delta(v_x, v_y) = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} (J_{t+1}(k, l) - J_t(k + v_x, l + v_y))^2$$

$$\min_{v_x, v_y} \Delta(v_x, v_y)$$

Таким образом, мы находим скорость перемещения, затем, исходя из темпа смены кадров, рассчитываем область копирования и переносим ее на дефектную компоненту.

Основная проблема этого метода - это расположение дефектной компоненты. Так, если она находится на границе раstra, то возможно движение, при котором этот метод не работает, а также тем смены кадров должен быть выше, чем длина компоненты взятая по вектору скорости деленная на $|v|$.

6. Восстановление изображения

Выбор метода восстановления изображения на ПЗС/КМО матрицы зависит от параметров оптической системы, технических характеристик матрицы, геометрии дефектных компонент, быстродействия вычислителя и т.д. Мы предлагаем технологию восстановления изображения, которая учитывает геометрию дефектных областей, особенности восстанавливаемого изображения.

6.1. Предварительный этап

Предварительный этап восстановления изображений выполняется один раз, перед началом эксплуатации ПЗС/КМО матрицы. На этом этапе решаются следующие задачи:

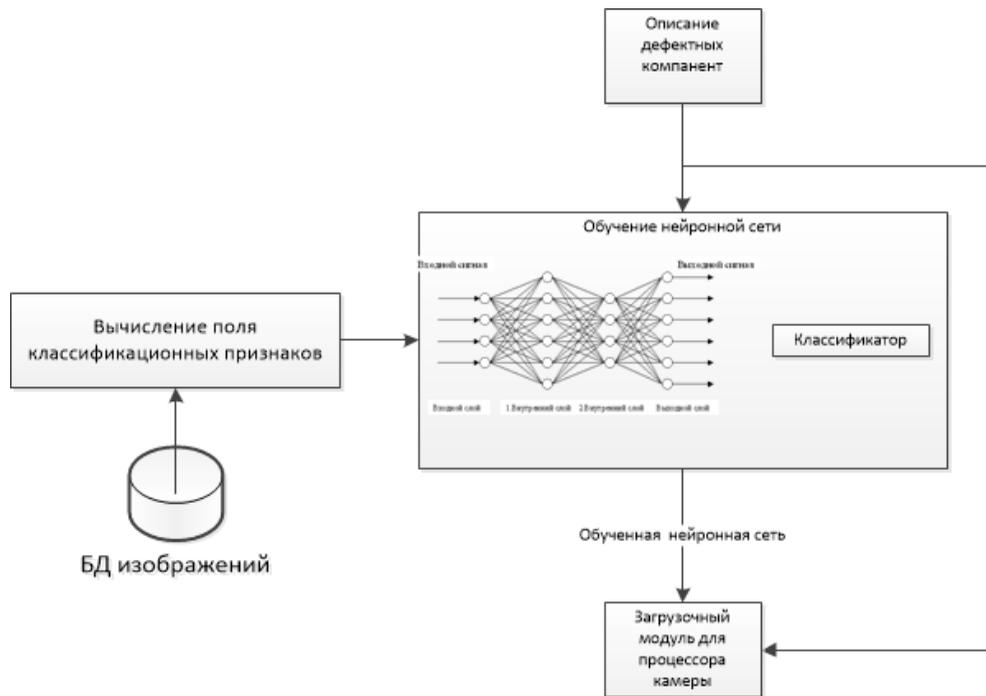


Рис. 10. Схема предварительного исследования ПЗС/КМО матрицы

- описание дефектных компонент (границы, координаты дефектной компоненты) матрицы;
- обучение нейронной сети на примерах.

Основные этапы предварительного исследования ПЗС/КМО представлены на Рис. 10.

Описание дефектной области состоит из списков дефектных компонент $k_1...k_m$ (координаты описывающих прямоугольников) и назначенных им расширенных окон $W_1(k_1)...W_k(k_m)$. Существует ряд алгоритмов, помогающих строить описание дефектных областей, но они не будут предметом нашего рассмотрения. Возможно «ручное» построение описание дефектной области.

Настройки классификатора изображений производятся с помощью обучения нейронной сети. Для обучения предварительно формируется эталонная база данных изображений. Каждый класс иерархии классов (п. 4.) должен быть представлен БД. В БД можно учесть условия эксплуатации ПЗС/КМО матрицы.

6.2. Машина восстановления изображений

Машины восстановления интегрирует все описанные выше алгоритмы в единую систему, обеспечивающую эффективное восстановление изображений.

Схема машины восстановления дефектного компонента

Структура машины восстановления представлена схемой на Рис. 11. Алгоритм работы следующий вначале восстанавливаются изолированные пиксели и линии дефектных пикселей. Затем восстанавливаются дефектные компоненты типа пятно. Каждое пятно k_i восстанавливается независимо от остальных и результаты восстановления k_i не зависят от результатов восстановления k_j .

Восстановление дефектных линий и изолированных точек

Из списка дефектных компонент $k_1...k_m$ выбираются изолированные точки и линии, к которым применяется алгоритм из п. 5.1. Затем восстанавливаются оставшиеся дефектные компоненты, которые имеют тип пятна.

Выделение и восстановление контуров

Пусть k - компонента типа пятно и W^- содержит набор контуров \underline{l}_i , прерванных дефектной компонентой k . Требуется правдоподобно восстановить непрерывность этих контуров или решить какие контура несопоставимы. В п. 5.2. предложен метод выделения и восстановления непрерывности для пары контуров. Рассмотрим случай, когда таких контуров более двух. Будем



Рис. 11. Машина восстановления компонент

считать, что отрезки контура распознаны. Необходимо в полученном списке отрезков контуров провести сопоставление. Возможны случаи, когда возникает неоднозначность выбора сопоставимых компонент предложенным ниже методом. Тогда ищем такие сопоставления, которые дают наименьший суммарный угол рассогласования при условии, что контура не пересекутся внутри дефектной области. Далее восстанавливаем непрерывность так, как описано в п. 5.2.

Вычисление поля классификационных признаков

Из всех классификационных признаков создаем поле признаков, которое позволяет с высокой вероятностью определять тип изображения согласно предложенной иерархии классов. Экстракторы признаков описаны в п. 3.

Классификация и выбор метода восстановления

Как говорилось выше, для получения удовлетворительных результатов необходимо исследовать изображение на W^- и, в зависимости от результатов исследования, видов дефектных областей, режима работы камеры, выбрать оптимальный алгоритм восстановления текстуры.

Если восстанавливается видеоряд и положение дефектной компоненты удовлетворяет условиям п. 5.5., то будем использовать Video Row модель.

Если восстанавливается отдельный кадр или положение дефектной компоненты не удовлетворяет условиям п. 5.5, то выполняется исследование W^- и классификация изображения при помощи нейронной сети согласно категориям представленным на Рис. 5. Затем из Табл. 2 выбирается методом восстановления текстуры.

Табл. 2. Соответствия классов и моделей восстановления

Класс изображения	Модель восстановления
Тайловая текстура	Синтаксическая модель восстановления
Фрактальная текстура	Синтаксическая модель восстановления
Текстура однородная	Стохастические текстуры на основе гистограммы
Неоднородная текстура	Модель использующая GLCM матрицу

Классификацией занимается нейронная сеть, которая является простым трехуровневым персептроном.

Генерация изображения

Сгенерированную текстуру накладываем на дефектную компоненту, затем поверх текстуры наклеиваются восстановленные контура.

Заключение

В статье разработан понятийный аппарат, позволяющий анализировать задачу восстановления дефектных компонент как математическую задачу. Разработана общая технологическая схема реабилитации ПЗС и КМОП матриц. Центральное место в этой технологии занимает машина восстановления изображения. Для отработки технологии и методов, используемых в машине восстановления, разработан стенд, состоящий из экстракторов признаков (C++ и Python), нейронной сети (TensorFlow) и моделей восстановления (C++). Был сформирован обучающий набор состоящий из отдельных изображений количеством около двух тысяч единиц и десять серий по сто последовательных кадров. На изображения накладывались три случайным образом сконфигурированные дефектные компоненты размера 50x50, что значительно выше технологической нормы дефектности матрицы. Классификации расширенных окон дефектных компонент была проведена экспертным путем. На этом наборе было произведено обучение нейронной сети. Полученная обученная сеть и модель восстановления обрабатывают несвязанные изображения в среднем за одну секунду. Для восстановления кадров в серии требуется в среднем менее чем 0,5 с. Очевидно, что скоростные результаты можно значительно улучшить, оптимизировав программное обеспечение. Выдержка современных камер находится в пределах от сотен микросекунд до нескольких секунд, поэтому предложенный подход реализуем. Качество восстановления оценивается, как процент не правильно восстановленных пикселей к общему числу пикселей на дефектной компоненте. Вычисления ведутся с учетом степени расхождения между пикселями. Предложены методы восстановления обеспечи-

вает достоверность на связанных изображениях порядка 82% на сериях кадров и порядка 97%.

Литература

1. Alfred V. Aho, Jeffrey D. Ullman. The Theory of Parsing, Translation, and Compiling, Volume 1: Parsing, 1972
2. Фурман Я.А. Кревецкий А.В. Передреп А.К. Роженцов А.А. Хафизов Р.Г. Егошина И.Л. Лехин А.Н. Введение в контурный анализ. - М.: ФИЗМАТЛИТ, 2005. - 561 с.
3. O. Acuna, Texture modeling using Gibbs distributions, Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing. 54(5), 210-222, (1992). 50.
4. Barnsley M. F. The science of fractal images – Springer-Verlag, NJ, 1988. – ISBN:0-587-96608-0
5. Cross G. C., Jain A. K. Markov random field texture models, IEEE Transactions on Pattern Analysis and Machine Intelligence. 5, (1985).
6. Chellappa R., Chatterjee S., Classification of textures using Gaussian Markov random fields, IEEE Transactions on Acoustics, Speech, and Signal Processing. ASSP {55(4), 959-965, (1985).
7. Fu K.S., Syntactic Pattern Recognition and Applications, Prentice-Hall, New Jersey, 1982.
8. Falconer K. J. Fractal geometry mathematical foundations and applications 2nd Ed. // IEEE, 2005. – 566 p. – ISBN 978-047084861677
9. Gabor D., Wilby W. P. and Woodcock R. (1960) A universal nonlinear filter, predictor and simulator which optimises itself by a learning process. *IEE Proc.*, 108: 422-438.
10. Goodfellow I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In NIPS, 2014.
11. Haralick R. M. Statistical and structural approaches to texture // Proceedings of the IEEE, 1979. – Vol 67, 5. – P. 786-804.
12. Haralick R. M. Textural Features for image classification // IEEE Transactions on systems, man and cybernetics, volume SMC-5. – IEEE, 1979. – 6. – P. 610-621. ISSN 0018- 9472.
13. Harte D. Multifractals. Theory and applications // Chapman & Hall/CRC, 2001. – 264 p. – ISBN 1-58488-154-2
14. Laws K.I. Textured Image Segmentation. PhD thesis, University of Southern California, LA, 1980.
15. Ljn Ljang, Ce Lju, Jng-Qing Ju ect. Real-Time Texture Synthesis by Patch-Based Sampling Microsoft Research China, Beijing.
16. O. L. Vovk. Evaluation of statistical features for texture classification // The Visnyk of the SSU, 2004. Vol. 71, no.12, p. 98-105.
17. Yong Xu, Hui Ji, Cornelia Fermuller. Viewpoint Invariant Texture Description Using Fractal Analysis. Int. J. Comp. Vis, 2009, 83, pp. 85-100.

Магазов Сергей Салимович. Доцент МГТУ Н.Э. Баумана К.Ф.-м.н. Количество печатных работ: 43 (в т.ч. 2 монографии). Область научных интересов: обработка сигнала, технологии искусственного интеллекта. E-mail: 2006mag@mail.ru

Image Recovery on Defective Pixels of a CMOS and CCD Arrays

S. S. Magazov

MGTU H.E. Bauman, Russia

Abstract The article investigates the problem of restoring a raster image on defective areas of CCD or CMOS matrices. The task of image restoration is divided into subtasks: restoration of contours and texture restoration. These problems are solved by a special image recovery machine, which uses image classification methods adapted to the task, a neural network, and image recovery methods. An original method of image restoration in the video series is proposed. The analysis of the computational complexity of the methods is fulfilled.

Keywords: texture generation, discrete function approximation, image statistics, Haralik and Laws parameters, Gabor filter, neural networks.

DOI 10.14357/20718632190303

References

1. Alfred V. Aho, Jeffrey D. Ullman. The Theory of Parsing, Translation, and Compiling, Volume 1: Parsing, 1972
2. Furman Ya.A. Krevetsky A.V. Peredreev A.K. Rozhentsov A.A. Khafizov R.G. Egoshina I.L. Leukhin A.N. Introduction to contour analysis. - M.: FIZMATLIT, 2005. –561 p.
3. O. Acuna, Texture modeling using Gibbs distributions, Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing. 54(5), 210{222, (1992). 50.
4. Barnsely M. F. The science of fractal images – Springer-Verlag, NJ, 1988. – JSBN:0-587-96608-0
5. Cross G. C., Jain A. K. Markov random field texture models, IEEE Transactions on Pattern Analysis and Machine intelligence. 5, (1985).
6. Chellappa R., Chatterjee S., Classification of textures using Gaussian Markov random fields, IEEE Transactions on Acoustics, Speech, and Signal Processing. ASSP{55(4), 959{965, (1985).
7. Fu K.S., Syntactic Pattern Recognition and Applications, Prentice-Hall, New Jersey, 1982.
8. Falconer K. J. Fractal geometry mathematical foundations and applications 2nd Ed. // IEEE, 2005. – 566 p. – JSBN 978-047084861677
9. Gabor D., Wilby W. P. and Woodcock R. (1960) A universal nonlinear filter, predictor and simulator which optimises itself by a learning process. *IEE Proc.*, 108: 422–438.
10. Goodfellow I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In NIPS, 2014.
11. Haralick R. M. Statistical and structural approaches to texture // Proceedings of the IEEE, 1979. – Vol 67, 5. – P. 786-804.
12. Haralick R. M. Textural Features for image classification //IEEE Transactions on systems, man and cybernetics, volume SMC-5. – IEEE, 1979. – 6. – P. 610-621. JSSN 0018- 9472.
13. Harte D. Multifractals. Theory and applications // Chapman & Hall/CRC, 2001. – 264 p. – JSBN 1-58488-154-2
14. Laws K.I. Textured Image Segmentation. PhD thesis, University of Southern California, LA, 1980.
15. Ljn Ljang, Ce Lju, Jng-Qjng Ju ect. Real-Time Texture Synthesis bj Patch-Based Sampling Microsoft Research China, Beijing
16. O. L. Vovk. Evaluation of statistical features for texture classification // The Visnyk of the SSU, 2004. Vol. 71, no. 12, p. 98–105.
17. Yong Xu, Hui Ji, Cornelia Fermuller. Viewpoint Invariant Texture Description Using Fractal Analysis. Int. J. Comp. Vis, 2009, 83, pp. 85-100.

Magazov S. S. PhD. MSTU H.E. Bauman, Department of IU-9 Moscow Russia, e-mail: 2006mag@mail.ru