

# МЕТОДИ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ



## Оптимальные непересекающиеся фильтры в обработке изображений

*А.А. ШУМЕЙКО, В.А. СМОРОДСКИЙ*

Днепродзержинский государственный технический университет

В статье предложена итерационная схема фильтрации изображений с использованием суперпозиции оптимальных непересекающихся фильтров.

В статті запропонована ітераційна схема фільтрації зображень з використанням суперпозиції оптимальних фільтрів, які не мають перетину.

The article suggests an image processing by iterative filtering scheme using superposition disjoint filters.

**Введение.** В задачах обработки цифровых данных важную роль занимает фильтрация исходной информации [1]. Результатом фильтрации является разделение обрабатываемой информации на «усредненную» часть, которая содержит основную информацию и «детализирующую» часть. Результаты фильтрации могут использоваться, как для изучения особенностей сигнала, так и для дальнейшего сжатия. Для последующего анализа особенностей сигнала наиболее важной является вторая – высокочастотная часть. С другой стороны, особенности визуального восприятия человеком таковы, что низкочастотные характеристики являются более информативными, а высокочастотные несут в себе достаточно большую часть избыточных данных, которыми можно пренебречь без особого ущерба для восприятия человеческим глазом.

Термин фильтр-дизайн определяет технологию выбора цифрового фильтра для тех или иных задач. Традиционно для фильтрации данных используется тригонометрическое преобразование Фурье [1]-[2]. В частности, широко используемый формат сжатия изображений JPEG для фильтрации каждой из трех цветовых компонент использует дискретное преобразование Фурье, разбивая изображения на квадраты со стороной в восемь пикселей. Результатом является 192 цифровых фильтра. Заметим, что преобразование Фурье применяется не к равноправным характеристикам цвета (традиционно используемых красного, зеленого и синего цветов), а к неравноправным – YCrCb, среди которых первая компонента является наиболее информативной и характеризует освещенность, вторая характеризует наличие теплых тонов, а третья – холодных. Таким образом, уже на этом этапе проходит фильтрация изображения.

Предложен метод оптимальной фильтрации, которые переводит исходные данные (в том числе и изображение) из совокупности равноправных характеристик в заданное множество неравноправных. Оптимальность метода состоит в том, что каждый последующий фильтр выбирает максимально возможную (в смысле среднеквадратичной нормы) информацию, что позволя-

ет разбить исходные данные на совокупность наиболее неравноправных характеристик.

**Постановка задачи и основной результат.** Вначале рассмотрим переход от трех равноправных цветовых характеристик к трем неравноправным по оптимальным формулам перехода.

Идея оптимальных непересекающихся фильтров состоит в том, что вначале подбираем первую компоненту  $C_{i,j}^0$  как решение задачи

$$\sum_{i,j} (R_{i,j} - \alpha^0 C_{i,j}^0)^2 + (G_{i,j} - \beta^0 C_{i,j}^0)^2 + (B_{i,j} - \gamma^0 C_{i,j}^0)^2 \rightarrow \min_{\alpha^0, \beta^0, \gamma^0}$$

при условии  $(\alpha^0)^2 + (\beta^0)^2 + (\gamma^0)^2 = 1$ .

После решения этой задачи находим ошибку, с которой исходные данные были восстановлены только одной компонентой  $C_{i,j}^0$ . К полученному массиву ошибки восстановления относимся как к исходным данным и находим вторую компоненту  $C_{i,j}^1$ , которая наилучшим образом восстанавливает уже ошибку. Далее этот процесс итерационно продолжаем.

Эта задача сводится к поиску собственных векторов [3]. Для случая трех компонентов решение этой задачи получается традиционными методами, но обобщая данный подход на любое наперед заданное число цветовых компонент, решение задачи традиционными способами становится не возможным. А.А.Лигуном [4] предложен итерационный метод, позволяющий достаточно быстро находить оптимальные фильтры для любого числа компонент. Вначале проиллюстрируем этот подход для трех цветовых компонент.

Для произвольного набора  $\alpha^0, \beta^0, \gamma^0$  такого, что

$$(\alpha^0)^2 + (\beta^0)^2 + (\gamma^0)^2 = 1.$$

Находим  $C_{i,j}^0$  из условия

$$\left(R_{i,j} - \alpha^0 C_{i,j}^0\right)^2 + \left(G_{i,j} - \beta^0 C_{i,j}^0\right)^2 + \left(B_{i,j} - \gamma^0 C_{i,j}^0\right)^2 \rightarrow \min \quad (1)$$

В результате получаем

$$C_{i,j}^0 = \alpha^0 R_{i,j} + \beta^0 G_{i,j} + \gamma^0 B_{i,j}.$$

Используя найденные значения  $C_{i,j}^0$ , исходя из условия

$$\sum_{i,j} \left(R_{i,j} - \alpha^0 C_{i,j}^0\right)^2 + \left(G_{i,j} - \beta^0 C_{i,j}^0\right)^2 + \left(B_{i,j} - \gamma^0 C_{i,j}^0\right)^2 \rightarrow \min_{\alpha^0, \beta^0, \gamma^0}$$

находим  $\alpha^0, \beta^0, \gamma^0$

$$\alpha^0 = \frac{\sum_{i,j} R_{i,j} C_{i,j}^0}{\sum_{i,j} (C_{i,j}^0)^2}, \quad \beta^0 = \frac{\sum_{i,j} G_{i,j} C_{i,j}^0}{\sum_{i,j} (C_{i,j}^0)^2}, \quad \gamma^0 = \frac{\sum_{i,j} B_{i,j} C_{i,j}^0}{\sum_{i,j} (C_{i,j}^0)^2}.$$

Проведем нормирование полученных значений  $\alpha^0, \beta^0, \gamma^0$  так, чтобы выполнялось условие

$$\left(\alpha^0\right)^2 + \left(\beta^0\right)^2 + \left(\gamma^0\right)^2 = 1.$$

В результате получим значения

$$\alpha^0 = \frac{\sum_{i,j} R_{i,j} C_{i,j}^0}{\varepsilon^0 \sum_{i,j} (C_{i,j}^0)^2}, \quad \beta^0 = \frac{\sum_{i,j} G_{i,j} C_{i,j}^0}{\varepsilon^0 \sum_{i,j} (C_{i,j}^0)^2}, \quad \gamma^0 = \frac{\sum_{i,j} B_{i,j} C_{i,j}^0}{\varepsilon^0 \sum_{i,j} (C_{i,j}^0)^2},$$

где

$$\varepsilon^0 = \sqrt{\left(\frac{\sum_{i,j} R_{i,j} C_{i,j}^0}{\sum_{i,j} (C_{i,j}^0)^2}\right)^2 + \left(\frac{\sum_{i,j} G_{i,j} C_{i,j}^0}{\sum_{i,j} (C_{i,j}^0)^2}\right)^2 + \left(\frac{\sum_{i,j} B_{i,j} C_{i,j}^0}{\sum_{i,j} (C_{i,j}^0)^2}\right)^2}.$$

Далее, используя в (1) новые значения  $\alpha^0, \beta^0, \gamma^0$  этот процесс повторяем несколько раз.

На следующем этапе находим ошибку, с которой  $C_{i,j}^0$  восстанавливает каждую цветовую компоненту  $R_{i,j}$ ,  $G_{i,j}$  и  $B_{i,j}$

$$\Delta R_{i,j} = R_{i,j} - \alpha^0 C_{i,j}^0,$$

$$\Delta G_{i,j} = G_{i,j} - \beta^0 C_{i,j}^0, \quad \Delta B_{i,j} = B_{i,j} - \gamma^0 C_{i,j}^0.$$

После этого, повторяем приведенный выше алгоритм - для произвольного набора  $\alpha^1, \beta^1, \gamma^1$  такого, что

$$\left(\alpha^1\right)^2 + \left(\beta^1\right)^2 + \left(\gamma^1\right)^2 = 1$$

из условия

$$\left(\Delta R_{i,j} - \alpha^1 C_{i,j}^1\right)^2 + \left(\Delta G_{i,j} - \beta^1 C_{i,j}^1\right)^2 + \left(\Delta B_{i,j} - \gamma^1 C_{i,j}^1\right)^2 \rightarrow \min$$

находим  $C_{i,j}^1$ . В результате получаем

$$C_{i,j}^1 = \alpha^1 \Delta R_{i,j} + \beta^1 \Delta G_{i,j} + \gamma^1 \Delta B_{i,j}.$$

Далее, аналогично предыдущему, рассмотрим задачу

$$\sum_{i,j} \left(\Delta R_{i,j} - \alpha^1 C_{i,j}^1\right)^2 + \left(\Delta G_{i,j} - \beta^1 C_{i,j}^1\right)^2 + \left(\Delta B_{i,j} - \gamma^1 C_{i,j}^1\right)^2 \rightarrow \min_{\alpha^1, \beta^1, \gamma^1}$$

находим решения этой экстремальной задачи  $\alpha^1, \beta^1, \gamma^1$

$$\alpha^1 = \frac{\sum_{i,j} \Delta R_{i,j} C_{i,j}^1}{\sum_{i,j} (C_{i,j}^1)^2}, \quad \beta^1 = \frac{\sum_{i,j} \Delta G_{i,j} C_{i,j}^1}{\sum_{i,j} (C_{i,j}^1)^2}, \quad \gamma^1 = \frac{\sum_{i,j} \Delta B_{i,j} C_{i,j}^1}{\sum_{i,j} (C_{i,j}^1)^2}$$

Проведем нормирование

$$\alpha^1 = \frac{\sum_{i,j} \Delta R_{i,j} C_{i,j}^1}{\varepsilon^1 \sum_{i,j} (C_{i,j}^1)^2}, \quad \beta^1 = \frac{\sum_{i,j} \Delta G_{i,j} C_{i,j}^1}{\varepsilon^1 \sum_{i,j} (C_{i,j}^1)^2}, \quad \gamma^1 = \frac{\sum_{i,j} \Delta B_{i,j} C_{i,j}^1}{\varepsilon^1 \sum_{i,j} (C_{i,j}^1)^2}$$

где

$$\varepsilon^1 = \sqrt{\left(\frac{\sum_{i,j} \Delta R_{i,j} C_{i,j}^1}{\sum_{i,j} (C_{i,j}^1)^2}\right)^2 + \left(\frac{\sum_{i,j} \Delta G_{i,j} C_{i,j}^1}{\sum_{i,j} (C_{i,j}^1)^2}\right)^2 + \left(\frac{\sum_{i,j} \Delta B_{i,j} C_{i,j}^1}{\sum_{i,j} (C_{i,j}^1)^2}\right)^2}.$$

Далее этот процесс повторяем несколько раз.

После этого находим

$$\Delta^2 R_{i,j} = \Delta R_{i,j} - \alpha^1 C_{i,j}^1,$$

$$\Delta^2 G_{i,j} = \Delta G_{i,j} - \beta^1 C_{i,j}^1, \quad \Delta^2 B_{i,j} = \Delta B_{i,j} - \gamma^1 C_{i,j}^1.$$

Повторяя аналогичные преобразования, получаем

$$C_{i,j}^2 = \alpha^2 \Delta^2 R_{i,j} + \beta^2 \Delta^2 G_{i,j} + \gamma^2 \Delta^2 B_{i,j}.$$

по которым находим

$$\alpha^2 = \frac{\sum_{i,j} \Delta^2 R_{i,j} C_{i,j}^2}{\varepsilon^2 \sum_{i,j} (C_{i,j}^2)^2}, \quad \beta^2 = \frac{\sum_{i,j} \Delta^2 G_{i,j} C_{i,j}^2}{\varepsilon^2 \sum_{i,j} (C_{i,j}^2)^2},$$

$$\gamma^2 = \frac{\sum_{i,j} \Delta^2 B_{i,j} C_{i,j}^2}{\varepsilon^2 \sum_{i,j} (C_{i,j}^2)^2}$$

где

$$\varepsilon^2 = \sqrt{\left(\frac{\sum_{i,j} \Delta^2 R_{i,j} C_{i,j}^2}{\sum_{i,j} (C_{i,j}^2)^2}\right)^2 + \left(\frac{\sum_{i,j} \Delta^2 G_{i,j} C_{i,j}^2}{\sum_{i,j} (C_{i,j}^2)^2}\right)^2 + \left(\frac{\sum_{i,j} \Delta^2 B_{i,j} C_{i,j}^2}{\sum_{i,j} (C_{i,j}^2)^2}\right)^2}.$$

Далее этот процесс повторяем, пока не получим, что величина

$$\sum_{i,j} \left(\Delta^2 R_{i,j} - \alpha^2 C_{i,j}^2\right)^2 + \left(\Delta^2 G_{i,j} - \beta^2 C_{i,j}^2\right)^2 + \left(\Delta^2 B_{i,j} - \gamma^2 C_{i,j}^2\right)^2$$

будет меньше наперед заданного числа.

В результате получаем формулы оптимального перехода

$$C_{i,j}^0 = \alpha^0 R_{i,j} + \beta^0 G_{i,j} + \gamma^0 B_{i,j},$$

$$C_{i,j}^1 = \alpha^1 \Delta R_{i,j} + \beta^1 \Delta G_{i,j} + \gamma^1 \Delta B_{i,j},$$

$$C_{i,j}^2 = \alpha^2 \Delta^2 R_{i,j} + \beta^2 \Delta^2 G_{i,j} + \gamma^2 \Delta^2 B_{i,j}.$$

В качестве примера приведем результат перевода тестового изображения "Lena" в три неравноправных характеристики.

Здесь используются три различных перехода, два из которых традиционные, и третий является оптимальным.

Первый RGB→YUV. Нормализованная матрица перехода имеет вид

$$A = \begin{pmatrix} 0.4472331059 & 0.8780128200 & 0.1705169701 \\ 0.7607972511 & -0.6370903810 & -0.1237068701 \\ -0.2705453249 & -0.5318884278 & 0.8024337527 \end{pmatrix}$$

то есть

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = A \begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{cases} 0.4472331059R + 0.8780128200G + 0.1705169701B \\ 0.7607972511R - 0.6370903810G - 0.1237068701B \\ -0.2705453249R - 0.5318884278G + 0.8024337527B \end{cases}$$

Отношение нормы коэффициентов  $Y$  к норме  $U$  равно 9.652348918, а к норме  $V$  равно 51.79721527.

Второй RGB→YCrCb. Нормализованная матрица перехода имеет вид

$$\begin{pmatrix} 0.4434103703 & 0.8705079844 & 0.2135488071 \\ 0.7608027105 & -0.6370809737 & -0.1237217368 \\ -0.2708386013 & -0.5316621929 & 0.8024847445 \end{pmatrix}$$

то есть

$$\begin{pmatrix} Y \\ Cr \\ Cb \end{pmatrix} = A \begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{cases} 0.4434103703R + 0.8705079844G + 0.2135488071B \\ 0.7608027105R - 0.6370809737G - 0.1237217368B \\ -0.2708386013R - 0.5316621929G + 0.8024847445B \end{cases}$$

Отношение нормы коэффициентов  $Y$  к норме  $Cr$  равно 9.961308244, а к норме  $Cb$  равно 53.382382.

Оптимальный переход RGB→ $C^0 C^1 C^2$ . В этом случае нормализованная матрица перехода имеет вид

$$\begin{pmatrix} 0.767785 & 0.45439 & 0.4517034 \\ -0.6164395 & 0.716085 & 0.3274513 \\ -0.174667 & -0.5298553 & 0.8299064 \end{pmatrix}$$

то есть

$$\begin{pmatrix} C^0 \\ C^1 \\ C^2 \end{pmatrix} = A \begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{cases} 0.767785R + 0.45439G + 0.4517034B \\ -0.6164395R + 0.716085G + 0.3274513B \\ -0.174667R - 0.5298553G + 0.8299064B \end{cases}$$

Отношение нормы коэффициентов  $C^0$  к норме  $C^1$  равно 72.80289228, а к норме  $C^2$  равно 182.3102984.

Отметим, что полученные фильтры являются ортогональными, а чем легко убедиться исходя из равенств

$$0.767785*(-0.6164395)+0.45439*0.716085+0.4517034*0.3274513=-0.2728*10^{-6};$$

$$0.767785*(-0.174667)+0.45439*(-0.5298553)+0.4517034*0.8299064=0.38902*10^{-5};$$

$$-0.6164395*(-0.174667)-0.5298553*0.716085+0.8299064*0.3274513=0.41352*10^{-5}.$$

Описанная технология легко обобщается для произвольного числа компонентов (каналов). Пусть дан исходный набор данных в виде  $N$  непересекающихся множеств  $X^k$ . Требуется найти набор непересекающихся множеств  $Y^k$  таких, что

$$\sum_{i,j} \sum_{k=1}^N (X_{i,j}^k - \alpha_k^0 Y_{i,j}^0)^2 \rightarrow \min$$

при условии  $\sum_{k=0}^N (\alpha_k^0)^2 = 1$ .

После этого найти ошибку восстановления каждого множества

$$\Delta X_{i,j}^k = X_{i,j}^k - \alpha_k^0 Y_{i,j}^0,$$

Используя полученный массив в качестве исходного, найти наилучшее восстановление, то есть найти решение задачи

$$\sum_{i,j} \sum_{k=1}^N (\Delta X_{i,j}^k - \alpha_k^1 Y_{i,j}^1)^2 \rightarrow \min$$

при условии  $\sum_{k=0}^N (\alpha_k^1)^2 = 1$ .

Вычисляя  $\Delta^2 X_{i,j}^k = \Delta X_{i,j}^k - \alpha_k^1 Y_{i,j}^1$  и опять, таки используя эти данные в качестве исходных, получаем итерационный метод построения оптимальных фильтров.

Приведем описание итерационного алгоритма построения оптимальных фильтров.

Выберем произвольным образом набор чисел

$\alpha_k^0$  таких, что  $\sum_{k=0}^N (\alpha_k^0)^2 = 1$  и положим

$$Y_{i,j}^0 = \alpha_k^0 X_{i,j}^k.$$

На следующем шаге вычислим  $\alpha_k^0 = \sum_{i,j} X_{i,j}^k Y_{i,j}^0$  для каждого  $k$ . Пронормируем полученные числа

$$\alpha_k^0 = \frac{\sum_{i,j} X_{i,j}^k Y_{i,j}^0}{\sqrt{\sum_{k=0}^N \left( \sum_{i,j} X_{i,j}^k Y_{i,j}^0 \right)^2}}$$
 и используем полученный

набор в качестве стартового – вычислим  $Y_{i,j}^0 = \alpha_k^0 X_{i,j}^k$ ,

по полученному множеству вновь найдем  $\alpha_k^0 = \sum_{i,j} X_{i,j}^k Y_{i,j}^0$ , проведем нормировку и повторим этот процесс заданное число раз. Полученный набор чисел  $\alpha_k^0$  будем называть нулевым фильтром, а множество  $Y_{i,j}^0$  — нулевым частотным доменом (или превью).

Следующим этапом является вычисление ошибки восстановления каждого множества

$$\Delta X_{i,j}^k = X_{i,j}^k - \alpha_k^0 Y_{i,j}^0.$$

Вновь выберем произвольным образом набор чисел  $\alpha_k^1$

таких, что  $\sum_{k=0}^N (\alpha_k^1)^2 = 1$  и положим

$$Y_{i,j}^1 = \alpha_k^1 \Delta X_{i,j}^k.$$

На следующем шаге вычислим  $\alpha_k^1 = \frac{\sum_{i,j} \Delta X_{i,j}^k Y_{i,j}^1}{\sum_{i,j} \Delta X_{i,j}^k}$  для каждого  $k$ . После нормирования получим числа

$$\alpha_k^1 = \frac{\sum_{i,j} \Delta X_{i,j}^k Y_{i,j}^1}{\sqrt{\sum_{k=0}^N \left( \sum_{i,j} \Delta X_{i,j}^k Y_{i,j}^1 \right)^2}}$$

набор в качестве стартового — вычислим  $Y_{i,j}^1 = \alpha_k^1 \Delta X_{i,j}^k$ , по полученному множеству вновь найдем  $\alpha_k^1 = \frac{\sum_{i,j} \Delta X_{i,j}^k Y_{i,j}^1}{\sum_{i,j} \Delta X_{i,j}^k}$ , пронормируем и повторим этот процесс заданное число раз. Полученный набор чисел  $\alpha_k^1$  будем называть первым фильтром, а множество  $Y_{i,j}^0$  — первым частотным доменом.

Далее вычислим

$$\Delta^2 X_{i,j}^k = \Delta X_{i,j}^k - \alpha_k^1 Y_{i,j}^1.$$

И построим итерационный процесс, вычисляя на каждом шаге  $\alpha_k^k$  и  $Y_{i,j}^k$  (то есть фильтр и соответствующий частотный домен).

Рассмотрим несколько применений оптимальной фильтрации.

Пусть данные определены на плоскости и задан паркет, плотно упаковывающий плоскость. Число пикселей, составляющих один элемент паркета равно  $N+1$ . Таким образом, данные естественно разбиваются на

$N+1$  непересекающихся множеств  $X^k, k = 0, 1, \dots, N$ .

При этом тенденция распределения частотных доменов такова, что «почти всегда» у каждого последующего домена среднеквадратичная норма меньше, чем у предыдущего. Но «почти всегда» не значит, что «всегда». Этот факт проиллюстрирован при фильтрации изображения `frimuge.bmp` в случае, когда паркет определен квадратами  $8 \times 8$  (рис. 1.1).

Восстановление системой оптимальных фильтров существенно зависит от того насколько плотная упаковка плоскости тем или иным паркетом. Этот факт проиллюстрирован на следующей диаграмме.

На следующей диаграмме по оси ОХ указано число используемых фильтров (и, соответственно, частотных доменов) для восстановления данных, а по оси ОУ — величина ошибки. Цвет линии соответствует виду паркета, на котором были построены оптимальные фильтры. В качестве примера использовалось тестовое изображение `frimuge.bmp` (рис. 1.2).

Таким образом, можно комбинировать различные методы фильтрации, например, традиционно вначале изображение фильтруется на три канала — YUV или YCrCb, потом к каждой из полученных цветных компонент применяется тот или иной метод разделения частотных характеристик. В качестве фильтра, как правило, используются либо фильтры Хаара, либо применяется дискретное преобразование Фурье. Понятно, что первичное разделение изображения на три цветных канала достаточно условно, можно использовать разделение на 12 и на 48 и др. число каналов. На следующем примере приведено качество фильтрации изображения `frimuge.bmp` различными методами восстановления (рис. 1.3).

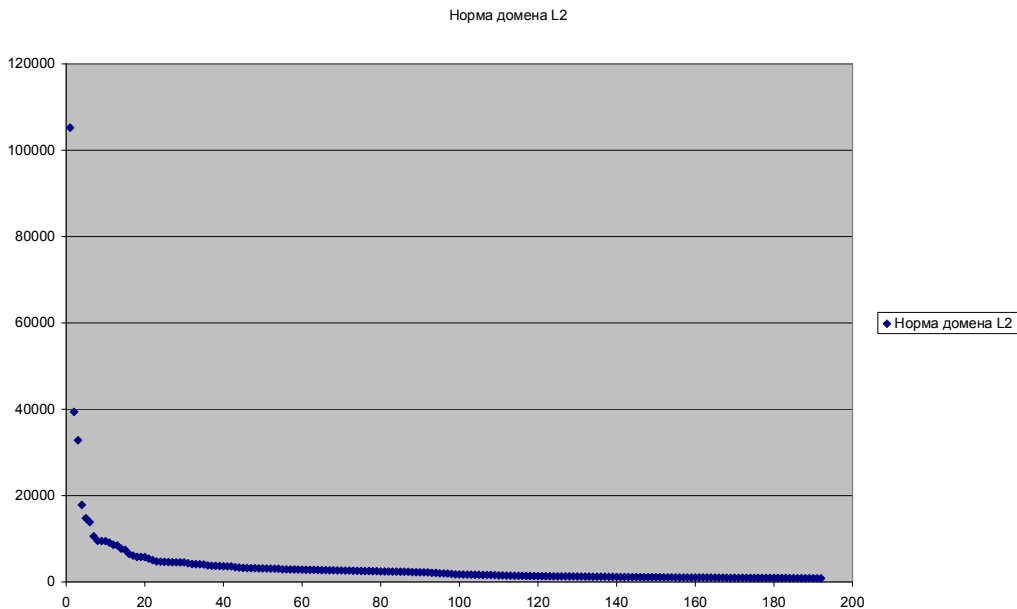


Рис. 1.1. Фильтрация изображения `frimuge.bmp` в случае, когда паркет определен квадратами  $8 \times 8$

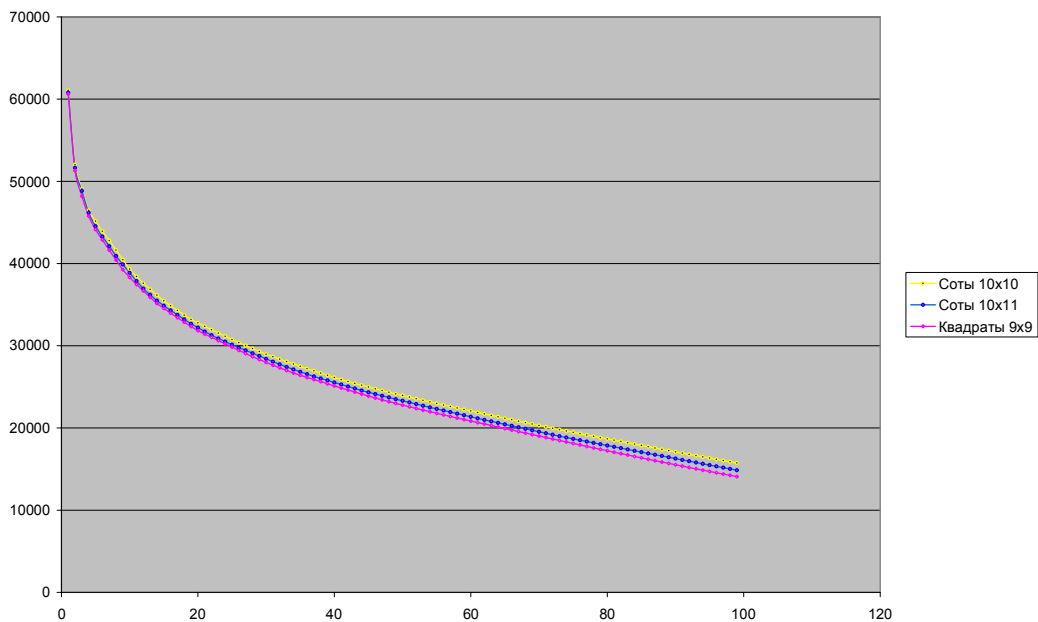


Рис. 1.2. Тестовое изображение frimyre.bmp

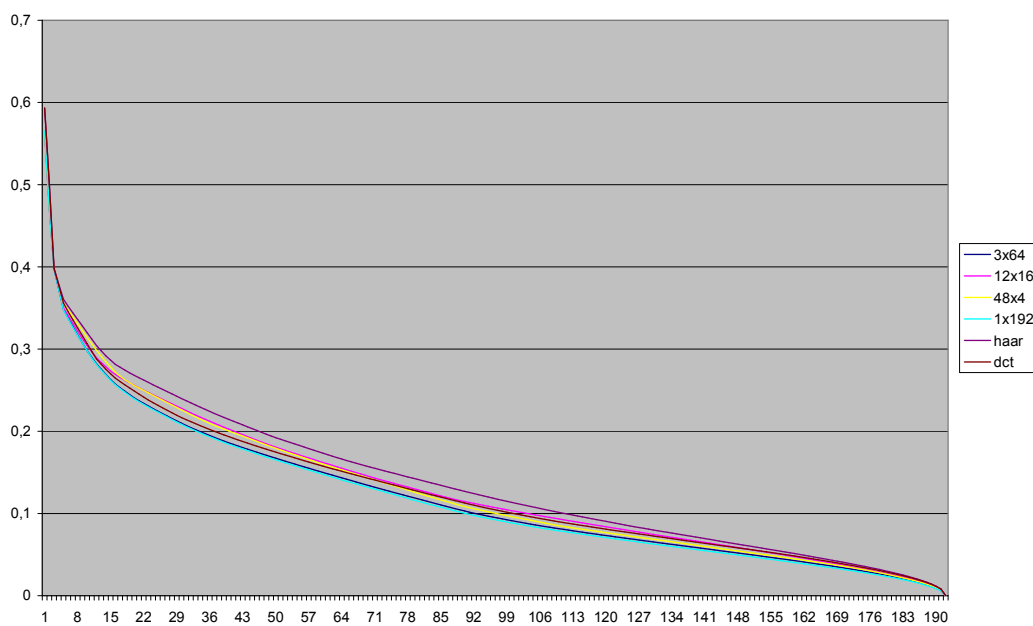


Рис. 1.3. Качество фильтрации изображения frimyre.bmp различными методами восстановления

Как видно из приведенного примера, любая схема, использующая вначале, разделение изображения на промежуточное число каналов, с последующей их фильтрацией, хуже, чем непосредственная оптимальная фильтрация на заданное число каналов (частотных доменов).

Таким образом, нами получено предельное значение, которое может быть реализовано ЛЮБЫМ линейным методом использующим непересекающиеся фильтры. Как следует из приведенной иллюстрации, результат использования оптимальных фильтров отличается от дискретного преобразования Фурье не так уж

намного, что делает безнадежным использования для фильтрации любого другого ортогонального преобразования, например, по полиномам Чебышева.

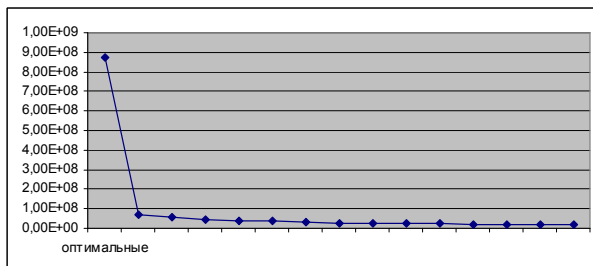
Важным применением ортогональных фильтров является исследование частотных особенностей данных. В отличие от традиционно используемых для этой цели преобразования Фурье по тригонометрическим системам функций, оптимальные фильтры более эффективно разделяют частотные характеристики, что полезно для широкого класса задач диагностики и подобное.

Использование оптимальной фильтрации для сжатия данных требует экономной записи не только

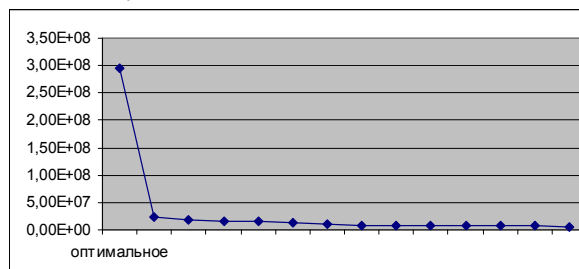
полученных частотных доменов, но и оптимальных фильтров. Так как полученные фильтры адаптивны к обрабатываемым данным, то они необходимы для корректного восстановления сигнала. Но даже при достаточно «жестком» квантовании фильтры отражают особенности обрабатываемого сигнала. Предложена методика сохранения фильтров, в основе которой лежит ортогонализация Шмидта. Как показали проведенные исследования, каждое число фильтра может быть записано в пять бит без существенной потери свойств оптимальной фильтрации. Еще раз отметим, что при исследовании сигнала задача компрессии фильтров вообще не стоит.

Отметим еще одно важное свойство оптимальной фильтрации. В силу того, что полученные фильтры являются ортогональными, их последующая фильтрация не является эффективной. Однако, если использовать коэффициенты полученных частотных доменов, взятые по модулю, то повторная фильтрация позволяет получить достаточно эффективное разделение на новую систему частотных доменов.

Например, если использовать шестнадцать полученных ранее частотных доменов тестового изображения `limage.bmp` с коэффициентами, взятыми по модулю, то получим следующую иллюстрацию

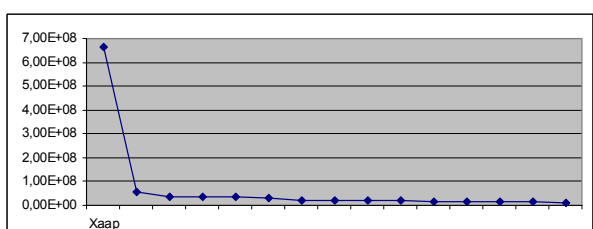


Более того, если мы возьмем из этих доменов со второго до шестнадцатого с коэффициентами по модулю и вновь проведем процедуру оптимальной фильтрации, то получим

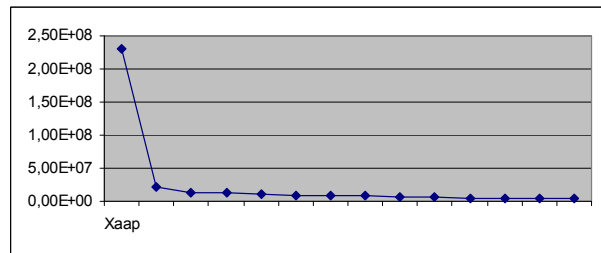


Это замечательное свойство оптимальной фильтрации сохраняется и для коэффициентов, полученных любым иным ортогональным фильтром.

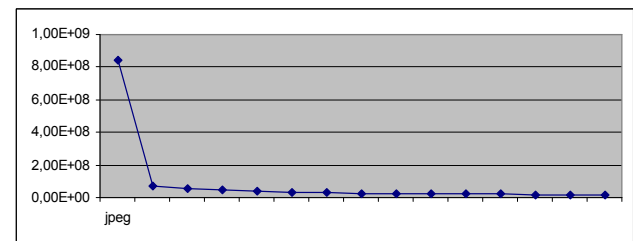
Например, для коэффициентов фильтрации методом Хаара, взятых по модулю и профильтрованных оптимальным образом получим следующую картинку



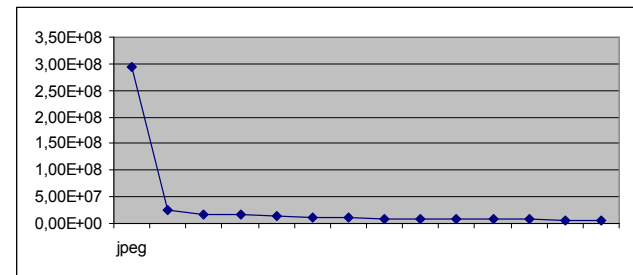
Последующая фильтрация коэффициентов, взятых по модулю, от второго до шестнадцатого домена приводит к следующему результату



Для дискретного преобразования Фурье имеем следующую картину



и



**Выводы.** Таким образом, проиллюстрирована возможность комбинированного использования как адаптивных, так и фиксированных фильтров. Этот факт позволяет в задачах сжатия уменьшить нагрузку на хранение значений оптимальных фильтров, сочетая использование на первом этапе, например, дискретного преобразования Фурье с последующей фильтрацией оптимальными фильтрами.

#### ЛИТЕРАТУРА

1. Гонсалес Р. Цифровая обработка изображений / Р.Гонсалес, Р.Вудс. – М.: Технофера, 2005. – 1070 с.
2. Лигун А.О. Комп'ютерна графіка (Обробка та стиск зображень): навч. посіб. / А.О.Лигун, О.О.Шумейко. – Д.: Біла К.О., 2010. – 114 с.
3. Cichocki A. Adaptive Blind Signal and Image Processing / A. Cichocki, S. Amari. – John Wiley&sons LTD, 2002. – 554 p.
4. Лигун А.А. О получении синтезированных изображений многозональной космической съемки / А.А. Лигун, А.А. Шумейко, Д.В. Тимошенко // Вісник Східноукраїнського національного університету імені Володимира Даля. – 2007. – №5(111). – С. 107-113.

пост. 08.02.2016