

Санкт-Петербургский государственный университет
Математико-механический факультет

Кафедра системного программирования

Улучшение качества видео с помощью вариационного подхода

Курсовая работа студента 344 группы
Пышновой Александры Витальевны

Научный руководитель

к.ф.-м.н., доцент А.Т. Вахитов

Санкт-Петербург
2013

Оглавление

Введение.....	3
1. Задача Superresolution.....	4
2. Постановка задачи.....	5
3. Алгоритм.....	6
4. V-VM3D.....	8
5. Реализация.....	10
6. Тестирование.....	11
Итоги.....	15
Список литературы.....	16

Введение

В наше время существует множество задач, для которых необходимы изображения или видео с высоким разрешением, например, в медицинских учреждениях, распознавания номеров автомобилей, лиц с видео-наблюдений. Для получения изображений или видео хорошего качества необходима и соответствующая техника. Но не всегда есть возможность воспользоваться специальным оборудованием. Во-первых, техника, позволяющая снимать четкие изображения и видео, является дорогостоящей. Во-вторых, если уже имеются снимки плохого качества, оборудование уже не поможет.

Для таких случаев был разработан отдельный раздел теории обработки сигналов - Superresolution. Superresolution - класс алгоритмических техник, решающих задачи повышения разрешения. Обычно используется для создания изображения высокого качества по набору изображений более низкого качества или улучшение качества всего видео.

В прошлом году студентами кафедры системного программирования Пасиченко В. и Землянским Ю. был придуман и реализован алгоритм для решения задачи Superresolution. Этот метод основан на применении фильтра BM3D и вариационного подхода. Во время тестирования данный алгоритм показал результаты, превышающие результаты методов, предложенных для сравнения в статьях [5], [6], что свидетельствует о его применимости для задач повышения разрешения. В связи с этим хотелось использовать вариационный подход для улучшения качества не только отдельного изображения. Поэтому целью этой курсовой работы стала реализация алгоритма, решающего задачу Superresolution для целого видео.

1. Задача Superresolution

Для начала необходимо описать модель наблюдений для задачи video

Superresolution:

$$y = Az + \sigma\varepsilon = BWD + \sigma\varepsilon$$

где z - видео хорошего качества, y - видео плохого качества, полученное с помощью матриц преобразований:

- B - размытие(blur). Этот эффект проявляется как артефакт во время фотографирования быстро движущихся в кадре объектов, и заключается в том, что объект смазывается; теряет свою резкость и выделяется на общем фоне
- D - уменьшение размерности (downsampling оператор). Этот оператор уменьшает размер изображения, удаляя каждую k строчку и каждый j столбец или, разделив все изображение на подматрицы размером $l \times l$, усреднить значения в них
- W - геометрическое искажение, означает изменение форм, начертаний элементов из-за движений и поворотов камеры
- $\sigma\varepsilon$ - аддитивный Гауссовский белый шум. Является изображением, накладываемым на кадр, полученный после матриц преобразований, где каждый пиксель - нормально распределенная случайная величина, с нулевым математическим ожиданием и конечной дисперсией σ^2 , ε - матрица размерности кадра видео плохого качества

Таким образом задача Superresolution для видео заключается в том, чтобы восстановить последовательность кадров z хорошего по известной последовательности кадров y плохого качества и известных матриц преобразований. Матрицы преобразований можно узнать с помощью существуют алгоритмов для распознавания параметров искажений, размытия.

2. Постановка задачи

Целью данной курсовой работы являются реализация алгоритма улучшения качества видео с помощью вариационного подхода и применения фильтра VBM3D. Так же необходимо протестировать и сравнить полученные результаты с результатами других программных средств (с помощью количественной метрики PSNR и визуально). Еще одна задача заключается в том, чтобы исследовать работу алгоритма в зависимости от параметров.

3.Алгоритм

Рассмотрим задачу Superresolution для нашего алгоритма. Поскольку видео - это последовательность кадров, то можно рассмотреть такую задачу для каждого отдельного кадра :

$$y = Bx + \sigma\epsilon,$$

В основном задача поиска изображения(отдельного кадра) сводится к задаче наименьших квадратов. И для большего соответствия реальному изображению вводят регуляризирующий функционал. Задача приобретает вид:

$$\frac{1}{2\sigma^2} \|y - Bx\|^2 + \tau \|\omega\|_{l_p} \rightarrow \min$$

Одной из идей придуманного алгоритма, использующего вариационный подход, было разделение минимизирующего функционала на два, поскольку так повышается эффективность. Задача приобретает такую интерпретацию:

$$L_{\text{inv}}(x, \omega) = \frac{1}{2\sigma^2} \|y - Bx\|^2 + \frac{1}{2\gamma_0} \|x - \Psi\omega\|^2,$$

$$L_{\text{den}}(x, \omega) = \tau \|\omega\|_{l_p} + \frac{1}{2} \|\Phi x - \omega\|^2.$$

Где первый функционал показывает насколько изображение x схоже с наблюдаемым y .

Второй функционал показывает насколько построенное изображение x является реальным изображением. σ, γ_0 - регуляризирующие параметры. Φ - спектральное преобразование изображения, Ψ - преобразование из спектра обратно в изображение.

Когда функционалы имеют такой вид, можно разделить представить задачу в виде системы :

$$\begin{cases} \mathbf{x}_{t+1} = \arg \min_{\mathbf{x}} L_{\text{inv}}(\mathbf{x}, \boldsymbol{\omega}_t) \\ \boldsymbol{\omega}_{t+1} = \arg \min_{\boldsymbol{\omega}} L_{\text{den}}(\mathbf{x}_{t+1}, \boldsymbol{\omega}) \end{cases}, \quad t = 0, 1, \dots$$

Данную задачу можно решить с помощью итеративного алгоритма, имеющего вид:

$$1. \quad \mathbf{x}_{t+1} = \Psi \boldsymbol{\omega}_t + \frac{k}{k+1} B^T (y - B \cdot \Psi \boldsymbol{\omega}_t) + (B^T B - I) \eta_t$$

$$2. \quad \boldsymbol{\omega}_{t+1} = Th_{\tau} \{ \Phi \mathbf{x}_{t+1} \}$$

\mathbf{x}_{t+1} - изображение, восстановленное из кадра плохого качества. $\boldsymbol{\omega}_{t+1}$ - его спектральное представление

4.V-BM3D

Одной из основных идей предложенного алгоритма было применение фильтра Block-Matching 3D(BM3D), разработанного научной группой Технологического Университета города Тампере под руководством профессоров К. Егиазаряна и В. Катковника. С помощью BM3D можно получить спектральное представление изображения, используемое для его восстановления.

Фильтр Video-Block-Matchin 3D(V-BM3D) имеет такое же назначение, только после его работы получается разряженное представление в спектральном базисе всего видео. Существует предположение, что реальные изображения имеют схожие блоки. Основная идея фильтра заключается в том, чтобы найти схожие блоки, сформировать в группы и применить к ним 3D преобразования.

На рис(1) изображена схема работы фильтра.

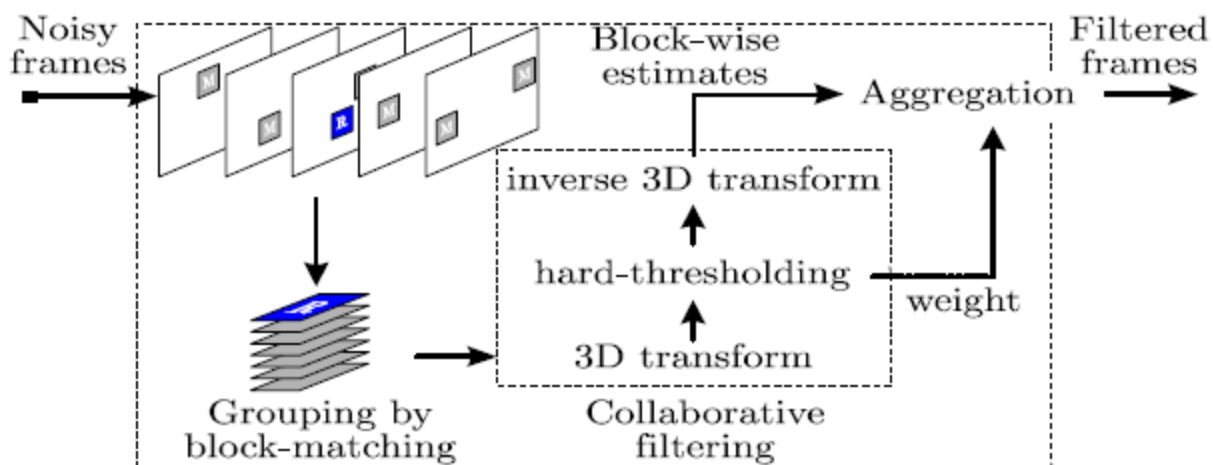


рис.1

Можно представить поэтапную работу фильтра:

1. Block-wise estimates. Каждый кадр входной последовательности рассматривается как набор блоков. Далее для каждого блока выполняется :

- Grouping by block-matching. Во всей последовательности кадров ищутся схожие блоки фиксированного размера. Далее они складываются в стек, образуя тем самым 3D массив (группу)

- Collaborative filtering. Далее к каждой группе применяется 3D преобразование: сначала применяется 2D преобразование (преобразование Фурье, дискретное преобразование Фурье, вейвлет преобразование и т.д.), далее применяется 1D межблочное преобразование (например, преобразование Хаара). После этого блоки возвращаются на исходные места
2. Aggregation. Далее спектры блоков для каждого кадра объединяются, если блоки накладываются, то значения усредняются.

Таким образом получается разреженное представление, необходимое для алгоритма.

5. Реализация

Мною был реализован алгоритм описанного подхода на Matlab. Исходники фильтра V-VM3D были представлены В.Катковником.

В самом алгоритме в качестве спектрального преобразования Ψ использовался фильтр V-VM3D. σ, γ_0 - исследовались и подбирались в зависимости от шума.

x_{t+1} - строился с помощью метода градиентного спуска.

Для преобразования видео хорошего качества в плохое использовался только оператор понижения расширения и добавление Гауссовского белого шума.

6.Тестирование

Целью курсовой работы являлась не только реализация алгоритма, но и тестирование программы и сравнение результатов с существующими решениями визуально и с помощью метрик. Для численного сравнения двух изображений в большинстве статей используется метрика PSNR (Peak signal-to-noise ratio). Этот термин обозначает соотношение между максимумом возможного значения сигнала и мощностью шума, искажающего значение сигнала. PSNR рассчитывается по формуле:

$$PSNR = 10 \log_{10} \left(\frac{MAX_I^2}{MSE} \right) = 20 \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right)$$

, где

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} |I(i, j) - K(i, j)|^2$$

, MAXI - максимальное значение принимаемое пикселем, I, K - два изображения размера $m \times n$, одно из которых является зашумленным приближением другого.

Для того, чтобы оценить насколько эффективно работает программа, было взято видео плохого качества, полученное искусственно из видео лучшего качества. Каждый кадр последовательности был уменьшен в 4 раза. Затем на него накладывался произвольный белый Гауссовский шум. Изначально предполагалось, что данный алгоритм будет хорошо работать для изображений с большим уровнем шума, поэтому на рис(2) и на рисунке(3) добавлен шум 15 и 20 соответственно. Видно, что на восстановленном изображении качество улучшилось, изображение стало более четким. Так же намного уменьшился уровень шума, что можно наблюдать не только визуально, но и по показанием метрики PSNR.



Исходное изображение



Изображение низкого качества



Восстановленное изображение

PSNR

изображение низкого качества -- 24.38

восстановленное изображение -- 35.25

рис.2

Уровень шума =15



Исходное изображение



Изображение низкого качества



Восстановленное

PSNR:

изображение низкого качества -- 21.98

восстановленное изображение -- 34.69

рис. 3

Уровень шума = 20

Для оценки эффективности так же необходимо сравнить работу алгоритма с уже существующими программами. Исходное видео уменьшили в 4 раза и прибавили шум с дисперсией 2. Реализованный метод оказался эффективнее большинства методов, описанных в статьях.



Исходное изображение



Изображение плохого качества



Восстановленное изображение

PSNR	
Non-Local	34.27
Lanczos Interpolation	31.25
Pixel Replication	28.99
Nearest neighbor	29.0
Наш алгоритм	33.7

Поставленные эксперименты показали, что реализованный алгоритм является эффективным в задаче повышения качества видео. Так же алгоритм с использованием вариационного подхода является устойчивым к шуму, что видно из рис(2) и рис(3).

Итоги

В процессе работы над поставленными задачами было выполнено:

- произведен обзор существующих алгоритмов для повышения качества видео
- реализован алгоритм Superresolution на основе вариационного подхода для видео
- алгоритм был протестирован для разных уровней шума
- произведено сравнение полученных результатов с результатами других программных средств

Список литературы

[1] Superresolution

[Электронный ресурс]URL: <http://en.wikipedia.org/wiki/Superresolution> (дата обращения : 27.11.2012)

[2] Danielyan A., Katkovnik V., Egiazarian K., “BM3D frames and variational image deblurring” //2011

[3] Katkovnik V., “Super-Resolution Imaging: Single Frame”

[4] Katkovnik V., Passichenko V., Vakhitov A., Zemlyanski Y., “Variational Super-Resolution” // 2012

[5] Danielyan A., Foi A., Katkovnik V., Egiazarian K., “Spatially adaptive Itering as regularization in inverse imaging: compressive sensing, super-resolution and upsampling” // 2010

[6] Protter M., Elad M., Takeda H., Milanfar P., ”Generalizing the Nonlocal-Means to Super-Resolution Reconstruction” // 2009

[7] Takeda H., Milanfar P., Protter M., Elad M., “Super-Resolution Without Explicit Subpixe Motion Estimation” // 2009

[8] Alessandro Foi

[Элетронный ресурс] URL: <http://www.cs.tut.fi/~foi/> (дата обращения : 15.12.2012)

[9] Peak signal-to-noise ratio

[Элетронный ресурс]URL: http://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio (дата обращения : 20.12.2012)

[10] Землянский Ю., “Использование вариационного подхода в задаче повышения разрешения изображения”, Дипломная работа // 2012