**Сравнение подходов обучения на базе словаря и MAP к проблеме повышения разрешения на примере изображений автомобильных номеров**

Улитин. А. А., студент кафедры системного программирования СПбГУ, [alexander.a.ulitin@gmail.com](mailto:alexander.a.ulitin@gmail.com)

Вахитов. А. Т., доцент кафедры системного программирования СПбГУ, [alexander.vakhitov@gmail.com](mailto:alexander.vakhitov@gmail.com)

**Аннотация**

В докладе дано сравнение двух известных подходов к проблеме повышения разрешения: на базе обучения словаря и maximum a posteriori оценивания с использованием некой заранее заданной функции плотности вероятности. Сравнение будет делаться на примере разработанной собственными силами базы данных автомобильных номеров.

# Введение

Цифровое изображение имеет конечное количество пикселей. Количество этих пикселей деленное на единицу площади называется разрешением цифрового изображения.

Для анализа и обработки изображений в большинстве случаев используются изображения высокого разрешения, так как они позволяют увидеть детали, которые не различимы, или плохо различимы на изображениях с низким разрешением. Многие задачи компьютерного зрения изначально полагаются на то, что изображение на входе в хорошем разрешение. С первого взгляда, задача увеличение разрешения чисто аппаратная – чтобы увеличить разрешение необходимо просто взять фотоаппарат с большей разрешающей способностью. Но часто уже имеется некоторая фотография или набор фотографий, и нет возможности повторно произвести съемку. Примером может послужить низкокачественные картинки с изображений камер наблюдения. В этом случае уместно использовать программные алгоритмы повышения разрешения. Эти алгоритмы используют некоторое дополнительное знание о изображении, для того, чтобы качественно повысить разрешение.

В этой работе рассмотрены два алгоритма для повышения разрешения на примере автомобильных номеров.

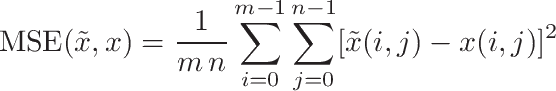
# Постановка задачи

Для рассуждений необходимо некоторым образом формально записать то, с чем имеем дело.

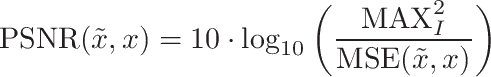
## Метрика PSNR

Для того, чтобы сравнивать несколько алгоритмов повышения разрешения необходимо ввести какую-нибудь количественную метрику. Чтобы иметь возможность посчитать метрику необходимо знать эталонное изображение, то есть такое изображение, которое должен выдать идеальный алгоритм повышения разрешения. Очевидно, что создать эталонный алгоритм невозможно - это отображение из $ \mathbb{N}^{n \cdot m } \to \mathbb{N}^{nk \cdot mk}, k \ge 2$. Знание эталонного изображения несколько выходит за рамки поставленной во введении задачи - в реальных условиях истинное изображение будет неизвестно. Для тестирования алгоритмов были использована стандартная модель получения изображений низкого разрешения из высокого.

Пусть $ x$- истинное изображение, $ \tilde{x}$- найденное изображение из изображений низкого разрешения. Посчитаем среднеквадратичную ошибку между пикселями этих изображений.



И выразим отношение $ \mathrm{MAX}_I$- максимальное значение яркости изображения к среднеквадратичной ошибке в децибелах $ \mathrm{PSNR}(\tilde{x},x)$.



Это и будет нашей метрикой.

## Интерполяция

Одним из наиболее известных способов повышения разрешения является интерполяция. Для любого изображения можно бесконечно повышать разрешения, просто добавляя дополнительные значения между пикселями исходного изображения. Однако, такое повышение разрешения не всегда дает хорошие результаты. Поскольку значение PSNR лишь показывает разницу между двумя изображениями, то нет возможности численно оценить насколько алгоритм Super-resolution хорошо справился с задачей. Но если сравнивать этот же алгоритм со значением полученные каким-либо методом интерполяции то наглядно видно, где алгоритм справляется хорошо, а где плохо. Более подроно про интерполяцию [1] [2]

## Изображения как векторы

Для возможности записывать все операции над изображением как матричное умножение, будем считать, что изображение высокого разрешения, которое мы хотим восстановить $ x$размера $ L_{1}N_{1}\times L_{2}N_{2}$записано в виде вектора в виде вектора $ x=[x_{1},x_{2},\dots,x_{N}]^{T}$, где $ N=L_{1}{N_{1}\times L_{2}N_{2}}$.

## Задача

$\displaystyle y_r = D H_r W_r x +\sigma_{k}\epsilon,~ ~ ~ 1 \leq r \leq m$

где:

* $ x$ оригинальное изображение
* $ y_r$ наблюдение $ r$
* $ D$ матрица понижение разрешения
* $ W_r$ матрица геометрического искажения
* $ H_r$ матрица размытия наблюдения $ r$
* $ \sigma_{k}\epsilon$ шум наблюдения $ r$
* $ m$ количество наблюдений

Задача найти:

$\displaystyle \tilde{x} = \underset{\hat{x}}{\operatorname{argmax}}~ PSNR(\hat{x},x)$

# Используемые алгоритмы

Для того, чтобы повысить разрешение автомобильного номера использовались два подхода: обучаемый [3] и интерполяционный с использованием регуляризации [4]. Этот выбор основывался исходя из природы изображения автомобильного номера.

## Обучаемый алгоритм на словарях

В статье [3] авторы предлагают использовать для задачи повышения разрешения одного изображения использовать пару связанных словарей $ D_x$и $ D_y$- словари содержащие патчи из пространства изображений с высоким и низким разрешением.

Алгоритм требует предварительного обучения нейронной сети. Для этого была создана база с изображениями автомобильных номеров. Использовав авторскую реализацию алгоритма была обучена пара словарей для восстановления изображений.

Обоснованием, для применения этого метода были следующие:

* Автомобильные номера содержат конечный набор патчей. Автомобильные номера очень похожи между собой. И явно видно, что между двумя изображениями автомобильных номеров есть общие кусочки изображения.
* Автомобильные номера имеют простую структуру - у них нет сложных переходов, большинство смены цветов это просто переход от белого к черному.

## Алгоритм с использованием регуляризации

В статье [4] авторы предлагают улучшение алгоритма MAP для задачи повышение изображения. Изображение моделируется как Марковская сеть и используется алгоритм шаговой оптимизации в сочетании с адаптивным регуляризатором для сохранения границ.

Общий вид алгоритма следующий:

1. Посчитать среднее от выровненных и билинейно интерполированных изображений $ X^{(0)}$. Будем называть полученные на этом шаге изображение «начальным приближением»
2. Положить $ \gamma = 2v$, где $ v$- максимальное значение градиента по осям $ x$ и $ y$ в начальном приближении $ X^{(0)}$
3. Do:
   1. $ X^{(n+1)} = X^{(n)} - \alpha\cdot\mathrm{grad}(X^{(N)}, \gamma) $
   2. $ n=n+1$
   3. If $ (\operatorname{norm}(X^{(n)}-X^{(n-1)}) < \epsilon)$  
      then $ \gamma^{(n)} = \max \{\gamma_{\mathrm{target}}, k\gamma^{(n-1)}\}$

Until $ (norm(X^{(n)}-X^{(n-1)} < \epsilon)$and $ (\gamma^{(n)} = \gamma_{\mathrm{target}})$

1. $ \hat{x} = X^{(n)}$

Где $ \alpha, \epsilon, k, \gamma{\mathrm target} $ параметры алгоритма. В нашей реализации мы использовали $ \alpha = 0.07, \epsilon=0.005, k=0.95, \gamma_{\mathrm target}=10$

$\displaystyle \mathrm{grad}(x, \gamma) = \frac{1}{\sigma^2} \sum^m_{r=1} W^T_R H^T_r D^T (D H_r W_r x-y_r) + \lambda \cdot G(x, \gamma)
$

Где $ \lambda$ экспериментально подобранные регуляризационный параметр и Градиент для точки $ (i,j)$задан следующей формулой:

\begin{displaymath}
\begin{array}{rcl}
G(i,j) & = & 2\left[x(i,j)-x(i,j-1)\ri...
... \exp\left(-[ x(i,j)-x(i+1,j) ]^2/\gamma \right)
\end{array}
\end{displaymath}

# Результаты

Для сравнения качества алгоритмов были выбраны несколько изображений из обучающей базы для тренировки словаря у первого алгоритма. Стоит заметить, что эти изображения были убраны из тренировочного набора.



Рисунок 1 Исходные изображения

|  |  |
| --- | --- |
| C:\Users\Sa\Dropbox\diplom\presentation\content\jpg\compare_result_sr1.jpg  Пример результа работы алгоритма с тренированными словарями | C:\Users\Sa\Dropbox\diplom\presentation\content\jpg\sr2_two_images.jpg  Пример результа работы алгоритма с использованием регуляризации |

|  |
| --- |
| C:\Users\Sa\Dropbox\diplom\presentation\content\jpg\pnsr_for_big_jpeg.jpg  Рисунок 2 Результаты алгоритма с тренированными словарями. На оси абсцисс отмечен номер изображения, из тестового набора. |
| C:\Users\Sa\Dropbox\diplom\presentation\content\jpg\compare_result_sr2.jpg  Рисунок 3 Результаты алгоритма с использованием регуляризации |

# Вывод

Как результат работы, можно заключить, что оба метода справляются с задачей повышения разрешения лучше, чем наивные интерполяционные подходы. Однако, результаты метода с использованием MAP намного лучше, чем результаты первого. Это можно объяснить тем, что второй алгоритм использует регуляризатор для выделения границ, что для большинства изображений будет давать не самые правильный результаты, но в случае автомобильных номеров это работает.

Полные результаты сравнения, реализации методов, словари для обучения и полные исходные коды на MatLab-е находятся в Git репозитории по адресу <https://github.com/ulex/sr-license-plates>

# Литература

|  |  |
| --- | --- |
| [1] | R. C. Gonzalez и E. Richard, «Woods, digital image processing,» *ed: Prentice Hall Press, ISBN 0-201-18075-8,* 2002. |
| [2] | R. Keys, «Cubic convolution interpolation for digital image processing,» *Acoustics, Speech and Signal Processing, IEEE Transactions on,* т. 29, № 6, pp. 1153-1160, 1981. |
| [3] | J. Yang, Z. Wang, Z. Lin, S. Cohen и T. Huang, «Coupled dictionary training for image super-resolution,» *Image Processing, IEEE Transactions on,* т. 21, № 8, pp. 3467-3478, 2012. |
| [4] | K. Suresh, G. M. Kumar и A. Rajagopalan, «Superresolution of license plates in real traffic videos,» *Intelligent Transportation Systems, IEEE Transactions on,* т. 8, № 2, pp. 321-331, 2007. |
| [5] | P. M. K. K. M. G. Park S. C., «Super-resolution image reconstruction: a technical overview,» *IEEE Signal Processing Magazine,* т. 20, pp. 21-36, 2003. |
| [6] | J. Tian и K.-K. Ma, «A survey on super-resolution imaging,» *Signal, Image and Video Processing,* т. 5, № 3, pp. 329-342, 2011. |