

ИЗВЛЕЧЕНИЕ ОСОБЫХ ТОЧЕК ИЗОБРАЖЕНИЯ ДЛЯ СОПРОВОЖДЕНИЯ ПЕШЕХОДОВ В РЕАЛЬНОМ ВРЕМЕНИ

А.С. Наздрюхин, И.Н. Храмцов, А.Н. Тушев

АлтГТУ им. Ползунова, г. Барнаул

Отслеживание объектов на кадрах видеозаписи в реальном времени может значительно упростить работу многих людей в наше время, будь то работа охранника или сотрудника государственной инспекции безопасности дорожного движения. В данной работе был описан и разработан алгоритм сопровождения объекта на видеокadre на примере пешеходов. Отслеживание пешеходов лишь частный случай, алгоритм может применяться для широкого спектра областей, как мирных, так и военных. В работе применяется принцип постоянной повторной идентификации и используются алгоритмы FAST и HOG, для поиска и идентификации объекта на кадре по ключевым точкам. Описаны преимущества разработанного алгоритма по сравнению с другими методами и алгоритмами, дающими тот же или схожий результат. Разработанный алгоритм показывает высокую скорость работы, что позволяет использовать его в реальном времени на маломощных встраиваемых системах. Описанный в данной работе алгоритм уступает алгоритмам с применением глубокого обучения в точности, но не требует большой базы данных пешеходов и больших вычислительных ресурсов, которые необходимы для обучения нейронной сети. Описанный в данной работе алгоритм работает быстрее своих аналогов, как с применением методов глубокого обучения, так и без. Так же приведен недостаток данного алгоритма, при исправлении которого можно серьезно повысить точность работы предложенного метода.

Ключевые слова: компьютерное зрение, трекинг, пешеходы, гистограмма направленных градиентов, FAST, принцип повторной идентификации.

ВВЕДЕНИЕ

Анализ содержимого видеокadre, обнаружение объектов – важные задачи компьютерного зрения, имеющие широкое применение. В частности, отслеживание объектов на некоторой последовательности изображений используется во многих системах компьютерного зрения, как военных (беспилотные летательные аппараты, тактические алгоритмы), так и мирных (медицинские приборы, автомобили и автопилоты, видеокамера на мобильном телефоне, системы Internet of Things).

В данной работе рассмотрены различные методы, помогающие сопровождать объект на видеокadre, однако основное внимание уделяется разработке алгоритма, способного отслеживать перемещение пешеходов с помощью нахождения и обработки ключевых точек в реальном времени по принципу постоянной повторной идентификации пешехода [1] на всех видеокadre. Такой анализ является особенно сложным, поскольку точки, необходимые для идентификации, могут иметь множество похожих точек на соседних видеокadre. Принцип постоянной повторной идентификации основан на предположении о том, что целевой объект доступен для идентификации не на всей последовательности видеокadre. Иначе говоря, алгоритму требуется повторно идентифицировать объект после его потери. Поэтому данный принцип также называют принципом реидентификации (англ. ReID).

Трекер - алгоритм, с помощью которого осуществляется автоматизация сопровождения

объекта на некотором видео. Сам процесс отслеживания называется трекингом. Трекер должен осуществлять работу следующим образом:

1. Трекер получает из видеопотока кадр (фрейм) и местоположение определенного объекта на данном фрейме.

2. Трекер получает следующие кадры, на которых он указывает местоположение целевого объекта.

Все существующие методы подразделяются на группы [2]:

1. Методы, использующие вероятностный подход. Данные методы анализируют некоторую последовательность кадров, на основе чего строится траектория движения и скорость объекта.

2. Методы, основанные на выделении некоторых характерных свойств объекта, таких как цвет, градиент, контур объекта и т.д.

Недостаток первой группы методов состоит в том, что при перекрытии целевого объекта другим объектом на видеокadre (например, отслеживаемый пешеход зашел за машину) область отслеживания «перескочит» с целевого объекта на загородивший его объект. Поскольку предполагается, что пешеходы перемещаются хаотично и часто закрывают друг друга, то прогнозирование траектории их движения даже при использовании гибридных подходов – является чрезвычайно сложной задачей. Поэтому выберем вторую группу методов.

Выделение характерных черт изображения в настоящее время осуществляется в соответствии с так называемой модульной парадигмой Марра [3]. Характерные черты, описывающие конкретный

объект, представляют, как отличительные атрибуты, с помощью которых можно поставить уникальную метку на объект или дифференцировать семейства объектов. В таком смысле особый интерес вызывают алгоритмы сегментации и выделения геометрической структуры. Однако алгоритмы сегментации в своем чистом виде более подходят для анализа отдельных изображений с высоким разрешением, нежели чем для видеопотока. Алгоритмы выделения геометрической структуры менее чувствительны к качеству изображения, и, следовательно, более подходят для решения нашей задачи. Алгоритмы выделения геометрической структуры анализируют:

1. Контур объекта.
2. Цветовые свойства (распределения).
3. Характерные (особые) точки.

Метод, основанный на анализе контура, не подходит для отслеживания пешеходов, поскольку их контуры сильно похожи между собой и не являются статическими (пешеходы могут махать руками и поворачиваться). Поэтому было решено скомбинировать два последних метода: для извлечения и последующего сравнения цветовых свойств в качестве примера был выбран алгоритм HOG (англ. Histogram of Oriented Gradients) [4], который применялся к множеству особых точек, полученных с помощью алгоритма FAST (англ. Features from Accelerated Segment Test) [5]. Для тестирования алгоритма было разработано приложение на базе библиотеки компьютерного зрения OpenCV на языке C++.

Следует заметить, что для данных методов существуют другие, гораздо более эффективные и точные алгоритмы, основанные на принципах глубокого обучения. Например, некоторые методы используют сверточную нейронную сеть для извлечения уникальных особенностей пешеходов [6,7]. Работа в [8] использует рекуррентную нейронную сеть для изучения изменений пространственных и временных характеристик пешеходов. Алгоритм обучения сети с помощью потери триплетов (англ. Triplet loss), предложенный в [9], выполняет кластеризацию объектов таким образом, чтобы идентификаторы, отвечающие за определенный объект, имели минимальное расстояние между собой по определенным метрикам, а экземпляры из разных классов идентификаторов были разделены как можно дальше. Однако такие алгоритмы требуют достаточно большую выборку пешеходов, что не всегда можно гарантировать, особенно в условиях реального времени. В то время как предложенный в данной работе алгоритм, требует для идентификации всего один видеокادر, что, в том числе, повышает скорость работы алгоритма, однако снижает его точность.

Анализировать объект возможно только тогда, когда известно его расположение на видеокadre.

Выделение объекта – отдельная задача, которая не рассматривается в данной работе. Методы нахождения и выделения объектов на изображении описаны в работах [10,11]. Мы, в свою очередь, будем считать, что расположение целевого объекта на изображении заранее известно. Пример выделенного объекта показан на рисунке 1. В многоугольник заключена область, где, предположительно, находится пешеход.

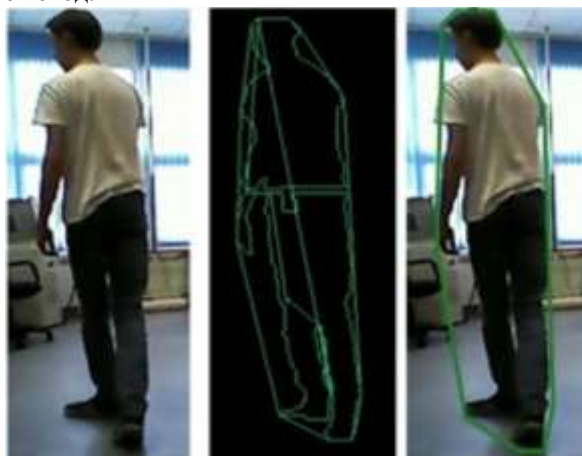


Рис. 1. Иллюстрация выделения пешехода

ОСНОВНАЯ ЧАСТЬ

Специфика разработки алгоритмов такова, что по мере увеличения сложности задачи, количество методов, подходящих для решения, уменьшается. Это так же верно и для данной задачи – для описания признаков особых точек, в основном, используется два подхода, первый из которых основан на обнаружении угловых точек, а второй – на работе с целыми областями изображения. Каждый подход должен определить локальные свойства некоторого объекта, в предположении о том, что такие свойства будут хорошо описывать объект в целом.

Особая точка должна соответствовать следующим требованиям [12,13]:

1. Отличимость – особая точка должна явно выделяться на фоне и быть отличимой в своей окрестности.
2. Инвариантность – определение особой точки должно быть независимо к аффинным преобразованиям.
3. Стабильность – определение особой точки должно быть устойчиво к шумам и ошибкам.
4. Уникальность – особая точка должна обладать глобальной уникальностью для улучшения различимости.
5. Интерпретируемость – особые точки должны определяться так, чтобы их можно было использовать для анализа соответствий и выявления интерпретируемой информации из изображения.

Угловая точка – это характерная (особая) точка, которая формируется при соприкосновении или

пересечении двух или нескольких линий. Также под угловой точкой можно понимать резкое изменение направления кривой. Второе определение позволяет нам использовать эффективный аппарат математического анализа для исследования детекторов угловых точек. Существуют множество детекторов угловых точек, основанных как на первом, так и на втором определении: вычисление матрицы Гессе [13], детектор Харриса [12,13], FAST и другие.

Приложение компьютерного зрения, представленное в данной работе, опирается на принцип встраиваемости – приложение должно работать не только на машинах, имеющих огромные производительные мощности, но и на «небольших» независимых системах, таких как микророботы или микрокомпьютеры. Такой принцип накладывает определенные временные ограничения и диктует постоянные стремления к повышению эффективности вычислений. Алгоритм FAST специально разработан с учетом данных ограничений и поэтому он является наиболее подходящим для задачи отслеживания в реальном времени.

В алгоритме FAST наличие угла определяется путем рассмотрения значений яркости точек изображения на дискретной окружности из 16 пикселей вокруг точки-кандидата p с интенсивностью L_0 (рис 2). Точка p является угловой, если на окружности существует не менее N пикселей, которые светлее $L_0 + t$ или темнее $L_0 - t$ (условие интенсивностей), где t – параметр алгоритма. Обычно $N = 8$. Однако в случае, если требуется определить угол с большой точностью, а не просто нерегулярный пиксель на окружности, значение N берется равным 11.

Для обеспечения эффективности вычислений интенсивность L_0 сравнивается с интенсивностями точек 1, 9, 5, 13 (рисунок 2) в указанном порядке. И лишь в том случае, если условие интенсивностей выполняется для каждой из указанных точек, рассматриваются и остальные точки на окружности.

В оригинальной работе, посвященной FAST, для большей оптимизации времени работы предлагается обучить дерево решений. В OpenCV используется альтернативный подход: детектор реализован с помощью обеспечения параллелизма при сравнении данных посредством SIMD-команд (англ. Single Instruction, Multiple Data) [13], что обеспечивает более высокую скорость вычислений, чем при использовании дерева решений. SIMD-команды поддерживаются большинством процессорных архитектур и современных видеоускорителей.

В алгоритме FAST также применяется подавление немаксимумов, чтобы получить более выразительное описание объекта. Для этого при вычислении интенсивностей для p , берется максимальная

разность T между L_0 и значениями изображения на окружности. Далее, если T меньше некоторого порогового уровня, то p не считается угловой точкой.

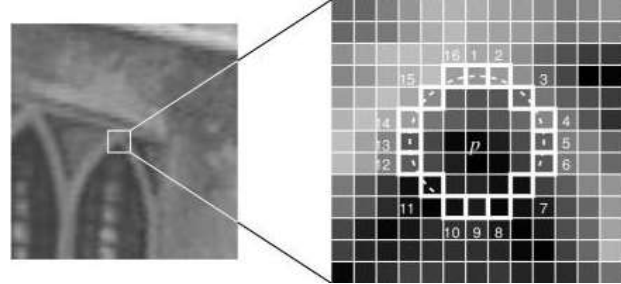


Рис. 2. Окружность с точкой-кандидатом p

Детектор FAST находит особые точки, но не дает способа для сравнения особых точек на различных изображениях. Для сравнения необходимо дополнить каждую точку дескриптором. Выделение особых точек на изображении мотивировано свойствами инвариантности и локализации признаков. Однако точки сами по себе обладают чрезвычайно плохой уникальностью. Для повышения уникальности характерных черт изображения обычно анализируются некоторая область, в центре которой находится особая точка.

Описание алгоритма HOG. HOG позволяет извлечь цветовые распределения изображения в специальный вектор – дескриптор. Особая точка в сочетании с дескриптором и будет являться нужным нам признаком. На рисунке 3 показана иллюстрация признака для некоторой области изображения. Алгоритм HOG основан на формировании гистограммы градиентов для изображения с разрешением 64x128 пикселей (рисунок 4).



Рис. 3. Иллюстрация HOG-дескриптора для некоторой части изображения

Гистограмма для каждого цветового канала строится по алгоритму:

1. Вычисление градиентов по x и y для каждого пикселя
2. Вычисление модуля и аргумента полученных градиентов.
3. Исходное изображение делится на ячейки размером 8x8 пикселей, для каждой ячейки формируется гистограмма из 9 ячеек; каждая ячейка

соответствует значению аргумента. В ячейки заносятся и суммируются модули градиентов (рис. 3).

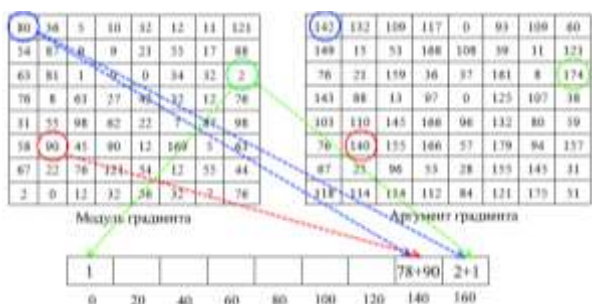


Рис. 4. Построение гистограммы для одного блока размером 8x8 пикселей

4. Значения градиентов зависят от интенсивности пикселей. Поэтому требуется нормализовать гистограмму, чтобы на нее не влияли вариации освещения. Нормализация идет по блокам размером 16x16 пикселей. Далее все нормализованные гистограммы объединяются посредством прямой суммы.

Для вычисления градиента используется оператор Собеля (для более точного вычисления градиентов можно использовать оператор Собеля-Шарра[14]), который применяется к сглаженным с помощью фильтра Гаусса изображением с нормировкой яркости. Оператор Собеля определяется по формулам:

$$G_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}; G_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$$

Тогда результат свертки изображения и матриц G_x и G_y будет являться соответствующими координатами вектора градиента. Далее по формулам можно получить G – модуль вектора градиента (1) и θ – угол направления градиента (2) для каждого пикселя.

$$G = \sqrt{G_x^2 + G_y^2} \quad (1)$$

$$\theta = \arctan\left(\frac{G_y}{G_x}\right) \quad (2)$$

Недостаток алгоритма НОГ заключается в том, что градиент хорошо показывает, где наблюдается резкое изменение интенсивностей (как правило, это контур объекта), но очень плохо учитывает цветовую информацию.

Общее описание использованного алгоритма. В соответствии с целями, описанными в предыдущих пунктах, автоматизация сопровождения пешеходов на видеокadre будет проводиться по следующему алгоритму (рис. 5):

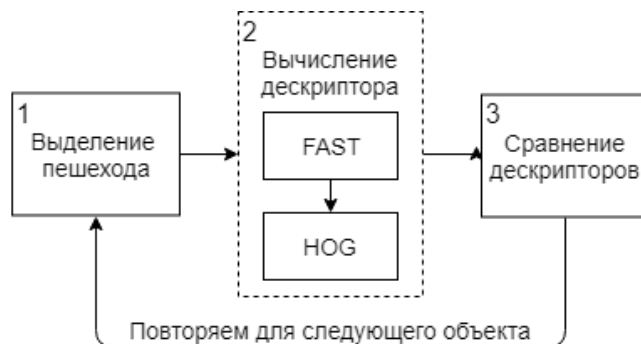


Рис. 5. Граф-схема алгоритма

Следует заметить, что дескриптор вычисляется для каждой области угловой точки; далее идет сравнение дескрипторов соответствующих областей (рис. 6).

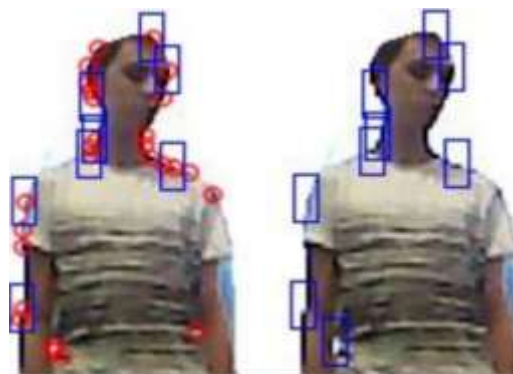


Рис. 6. Отображение некоторых областей, для которых вычисляется дескриптор

ЗАКЛЮЧЕНИЕ

Алгоритм был реализован на базе библиотеки компьютерного зрения OpenCV. Результаты показаны на рисунках 7 и 8 (окружностями отмечены угловые точки, цифрами – их количество). На рисунке 7 обнаружено 39 похожих точек для двух кадров, на рисунке 8 – всего 7 похожих точек.

Алгоритм уступает нейронным сетям глубокого обучения в точности, но его преимуществом является отсутствие необходимости обучения модели, а следовательно, нам не нужны огромные базы данных пешеходов для обучающих выборок и вычислительные ресурсы для непосредственного обучения. Что касается других методов выделения особых точек и получения их дескрипторов, то время их работы больше, чем время работы данного алгоритма, а в данной задаче это играет важную роль.



Рис. 7. Сравнение изображений пешехода на соседних кадрах



Рис. 8. Сравнение изображений пешехода на случайных кадрах

Разработанный алгоритм сопровождения пешеходов является достаточно эффективным, однако неустойчивым к резкому появлению новых угловых точек, а значит, корректный трекинг возможен только для пешеходов, которых видно в полный рост. Также видно, что алгоритм FAST находит угловые точки на контуре пешехода (рис. 7, 8), почти полностью игнорируя центральную часть

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Benenson, R., Omran, M., Hosang, J. and Schiele, B., “Ten Years of Pedestrian Detection, What Have We Learned?”, LNCS: Computer Vision – ECCV 2014 Workshops, vol.8926, pp.613-627, 2014.

2. Alper Yilmaz, Omar Javed and Mubarak Shah, “Object Tracking: A Survey,” ACM Journal of Computing Surveys, Vol. 38, No. 4, pp. Dec. 2006.

3. Обработка и анализ цифровых изображений с примерами на LabVIEW и IMAQ Vision / Ю.В. Визильтер и др. - Москва: ДМК Пресс, 2012.

4. Viola, P., Jones, M.J. and Snow, D., “Detecting pedestrians using patterns of motion and appearance,” In: Proc of the 9th International Conf. of Computer Vision, Nice, vol. 1, pp. 734–741, 2003.

5. Rosten, E. and Drummond, T., “Machine learning for high speed corner detection,” 9th European Conference on Computer Vision, vol. 1, pp. 430–443, 2006.

6. Krizhevsky, A., Sutskever, I., and Hinton, G.E., “ImageNet Classification with Deep Convolutional Neural Networks,” NIPS, 2012.

7. Girshick, R., Donahue, J., Darrell, T., and Malik, J., “Rich feature hierarchies for accurate object detection and semantic segmentation,” CVPR, 2014.

8. McLaughlin, N, del Rincon, J.L., and Miller, P. C., “Re-current convolutional network for video-based person reidentification,” CVPR, 2016.

9. Cheng, D., et al, “Person Re-identification by Multi-Channel Parts-Based CNN with Improved Triplet Loss Function,” CVPR, 2016. [Online]. Available: <http://ieeexplore.ieee.org/document/7780518/>.

10. Liang, X. et al, “Human Parsing with Contextualized Convolutional Neural Network”. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 39. 2016.

11. Hao Zhu, et al, “Detailed Human Shape Estimation from a Single Image by Hierarchical Mesh Deformation,” arXiv:1904.10506, 2019.

12. Gonzalez, R., Woods, R., Digital Image Processing, 4th edn. New York: Pearson, pp. 868-903, 2018.

13. Klette, R., Concise Computer Vision, London: Springer, pp. 396-407, 2014.

14. Jähne, B., Scharf, H., “Principles of Filter Design,” Handbook of Computer Vision and Applications. Vol 2. pp.125-151, 1999.

Наздрюхин Александр Сергеевич – студент кафедры ИВТиИБ, ФГБОУ ВПО АлтГТУ, тел. +7-952-007-27-99, e-mail: a.nazdryukhin@gmail.com.

Храмцов Игорь Николаевич – студент кафедры ИВТиИБ, ФГБОУ ВПО АлтГТУ, тел. +7-923-647-19-21, e-mail: igorxramcov@yandex.ru.

Тушев Александр Николаевич – доцент кафедры ИВТиИБ, ФГБОУ ВПО АлтГТУ, кандидат технических наук, тел. +7-913-081-62-85, e-mail: tushev51@mail.ru.

FEATURE POINT EXTRACTION FOR REAL-TIME PEDESTRIAN TRACKING

A.S. Nazdryukhin, I.N. Khramtsov, A.N. Tushev
Altai state technical University, Barnaul

The tracking of objects in real time video can significantly facilitates works of many people, such as, for example, security guards or road policemen. This paper presents algorithm for tracking pedestrians as some objects in video frames. Pedestrians are only one example the using of algorithm, but proposed algorithm can be useful in many civil and military areas. We use the principle of repeatedly identification of objects, which includes algorithms FAST and HOG for searching object in frame by key points. It is shown in work the advantage of developed algorithm compared to methods and algorithms with same or similar results. Our algorithm has high performance and can be used in low power built-in real time systems. In spite that developed algorithm in accuracy is below methods of deep learning, it is not require big data base of pedestrians and vast amount of processing power for training of neural nets. Algorithm presented in this paper have higher speed in comparison with deep learning methods and other methods, proposed earlier by several authors. We show some deficiencies of algorithm, its correction in future will significant increase the accuracy of method.

Keywords: computer vision, tracking, pedestrian, histogram of oriented gradients, FAST, Re-Identification.

REFERENCES

1. Benenson, R., Omran, M., Hosang, J. and Schiele, B., "Ten Years of Pedestrian Detection, What Have We Learned?", LNCS: Computer Vision – ECCV 2014 Workshops, vol.8926, pp.613-627, 2014.
2. Alper Yilmaz, Omar Javed and Mubarak Shah, "Object Tracking: A Survey," ACM Journal of Computing Surveys, Vol. 38, No. 4, pp. Dec. 2006.
3. Processing and analysis of digital images with examples on LabVIEW and IMAQ Visio / Yu.V. Visilter et al. - M: DMK Press, 2012.
4. Viola, P., Jones, M.J. and Snow, D., "Detecting pedestrians using patterns of motion and appearance," In: Proc of the 9th International Conf. of Computer Vision, Nice, vol. 1, pp. 734–741, 2003.
5. Rosten, E. and Drummond, T., "Machine learning for high speed corner detection," 9th Euproean Conference on Computer Vision, vol. 1, pp. 430–443, 2006.
6. Krizhevsky, A., Sutskever, I., and Hinton, G.E., "ImageNet Classification with Deep Convolutional Neural Networks," NIPS, 2012.
7. Girshick, R., Donahue, J., Darrell, T., and Malik, J., "Rich feature hierarchies for accurate object detection and semantic segmentation," CVPR, 2014.
8. McLaughlin, N, del Rincon, J.L., and Miller, P. C., "Re-current convolutional network for video-based person reidentification," CVPR, 2016.
9. Cheng, D., et al, "Person Re-identification by Multi-Channel Parts-Based CNN with Improved Triplet Loss Function," CVPR, 2016. [Online]. Available: <http://ieeexplore.ieee.org/document/7780518/>.
10. Liang, X. et al, "Human Parsing with Contextualized Convolutional Neural Network". IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 39. 2016.
11. Hao Zhu, et al, "Detailed Human Shape Estimation from a Single Image by Hierarchical Mesh Deformation," arXiv:1904.10506, 2019.
12. Gonzalez, R., Woods, R., Digital Image Processing, 4th edn. New York: Pearson, pp. 868-903, 2018.
13. Klette, R., Conscise Computer Vision, London: Springer, pp. 396-407, 2014.
14. Jähne, B., Schar, H., "Principles of Filter Design," Handbook of Computer Vision and Applications. Vol 2. pp.125-151, 1999.

Nazdryukhin Alexandr Sergeevich – student at the chair of Informatics, Computer engineering and Information security, Altai state technical University, +7-952-007-27-99, e-mail: a.nazdryukhin@gmail.com.

Khramtsov Igor Nikolaevich – student at the chair of Informatics, Computer engineering and Information security t, Altai state technical University, +7-923-647-19-21, e-mail: igorxramcov@yandex.ru.

Tushev Alexandr Nikolaevich– associate professor at the chair of Informatics, Computer engineering and Information security, Candidate of Engineering Science, Altai state technical University, +7-913-081-62-85, e-mail: tushev51@mail.ru.