

Санкт–Петербургский Государственный Университет

Самарин Алексей Владимирович

Выпускная Квалификационная Работа

Легковесная модель для автоматического улучшения
изображений

Уровень образования: аспирантура

Направление 05.13.11 «Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей»
Основная образовательная программа МК.3019.2017 «Информатика»

Научный руководитель:
профессор, кафедра информатики,
д.ф. - м.н. Косовская Татьяна Матвеевна

Рецензент:
доцент, кафедра телевидения и видеотехники,
к. т. н. Мотыко Александр Александрович

Санкт-Петербург
2021 г.

Saint Petersburg State University

Aleksei Samarin

PhD student thesis

Lightweight model for automatic image enhancement

Education level: postgraduate studies

Speciality 05.13.11 «Mathematical and software support of computers,
systems and computer networks»

Programme MK.3019.2017 «Informatics»

Scientific supervisor:

Doctor of Physics and Mathematics, professor. Tatiana Kosovskaya

Reviewer:

Ph.D., docent. Alexandr Motyko

Saint Petersburg

2021

Содержание

Введение	4
Постановка задачи	8
Обзор литературы	9
Глава 1. Архитектура LFIEM	14
1.1. Генератор параметров	15
1.2. Преобразования изображений	16
1.3. Функция потерь с регуляризацией согласованности	21
Глава 2. Постановка экспериментов	22
2.1. Набор данных	22
2.2. Метрики качества	23
2.3. Детали реализации	24
Глава 3. Анализ полученных результатов	25
3.1. Анализ фильтров	26
3.2. Влияние регуляризации согласованности	27
3.3. Сравнительный анализ с другими методами	28
Глава 4. Интеграции LFIEM в комбинированные архитектуры клас- сификации изображений	30
4.1. Задача классификации изображений в сложных условиях	30
4.2. Интеграция LFIEM в комбинированный классификатор удо- стоверений личности	31
4.3. Интеграция LFIEM в комбинированный классификатор фо- тографий фасадов коммерческих зданий	32
4.4. Анализ результатов интеграции	32
Заключение	33
Список литературы	34

Введение

В настоящее время, функционал для ретуши фотографий интегрируются во все большее число мобильных приложений. Многие современные подходы, основанные на механизмах машинного обучения, улучшают цветовую гамму изображений с помощью тяжеловесных моделей на основе сверточных нейронных сетей, где результат получается непосредственно из выходных данных нейронной сети. Подобные методы могут приводить к появлению артефактов в результирующих изображениях. Более того, архитектура подобных моделей, как правило сложно интерпретируема, требует длительного и трудоемкого процесса обучения, а также занимает много места в памяти устройств и требует много времени при обработке фотографий.

В последние годы фотография стала неотъемлемой частью жизни многих людей, и объем создаваемых изображений неуклонно растет. Снимки самых разнообразных объектов выполняются в различных условиях окружающей среды с использованием различных устройств. Это может привести к появлению артефактов на изображениях, ухудшающих визуальное восприятие фотографий. Поскольку качественная ручная постобработка изображений требует от фотографа значительных усилий и специальных навыков, автоматическое улучшение изображений (см. Рис. 1) становится все более актуальным.

Задача автоматического улучшения изображений еще не решена в общем виде из-за большого количества факторов, таких как яркость, контраст, локальные артефакты изображения, размытость и другие аспекты, влияющие на восприятие изображения человеческим глазом. В настоящее время существует несколько типов подходов к решению задачи автоматического улучшения изображений.



Рис. 1: Пример автоматического улучшения цветовой гаммы изображения с помощью модели, разработанной в ходе выполнения данной работы (LFIEM [1]). Оригинальное изображение получено из набора данных Adobe FiveK dataset. (Слева) расположено исходное изображение. (Справа) расположено изображение, полученное с помощью LFIEM [1] из исходного. LFIEM [1] не порождает артефактов и показывает хорошие результаты на изображениях с высокой степенью детализации.

Существует категория подходов, которые, прогнозируют параметры фильтров для последовательной обработки изображений [2, 3]. Однако, отсутствие всестороннего сравнительного анализа изображений порождает вопрос о наиболее оптимальном стеке фильтров для улучшения изображений. Более того, несмотря на то, что обработка изображений непосредственно на мобильных устройствах является весьма прогрессивным подходом, поскольку не требует передачи данных или дополнительных зависимостей на сервер, чем облегчает обработку информации в реальном времени, подавляющее число методов улучшения цветовой гаммы изображений не подходят для использования на мобильных устройствах.

Как было отмечено ранее, многие подходы основаны на получении результата улучшения изображения непосредственно из выходов нейронной сети [4–6]. Недостатком таких методов является то, что они могут привести к неестественному искажению результирующего изображения, которое обычно вызвано разницей в разрешении между обрабатываемым изображением и выборками из обучающего набора. Кроме того, большинство "чистых" моделей на основе только CNN не могут быть развернуты на мобильных платформах, поскольку такие модели довольно тяжеловесны.

Для решения вышеупомянутых проблем, в рамках данной работы создана облегченная модель автоматического улучшения цветовой гаммы изображений (Lightweight Filter based Image Enhancement Model - LFIEM [1]), которая подходит для использования на мобильных устройствах. Кроме того, в рамках данного исследования, предлагается комплексный сравнительный анализ фильтров для построения наиболее оптимального метода цветокоррекции и проведения сравнительного анализа представленного решения с другими моделями. Таким образом, в рамках данной работы произведено исследование применения подхода к коррекции цветовой гаммы изображений на основе фильтров, решающего описанные выше проблемы. При разработке данного решения был сделан акцент на создании легковесного метода, подходящего для использования на мобильных устройствах. Значительное повышение производительности было достигнуто за счет адаптации и применения регуляризации согласованности, используемой в обучении с частичным привлечением учителя. Предлагаемая модель может быть использована на мобильных устройствах и обеспечивает конкурентоспособные результаты по сравнению с известными подходами.

Таким образом, разработанная модель (LFIEM [1]) обладает следующими характеристиками:

1. Предложенная модель легковесна и может использоваться на мобильных устройствах. По сравнению с моделями с аналогичными значениями метрик качества на наборе данных MIT Adobe FiveK [7], предложенная модель содержит значительно меньше весов. Более того, небольшой размер предлагаемой модели приводит к высокой скоро-

сти работы, что расширяет возможности развертывания.

2. Разработанное решение не склонно к генерации артефактов, в отличие от большинства моделей на основе CNN [4, 6], которые получают результирующее изображение непосредственно из выходов сверточной нейросетевой архитектуры.
3. Так как для разработки предложенной модели было проведено обширное исследование обучаемых преобразований изображений, среди многочисленных аналогов, была выбрана оптимальная стратегия улучшения изображений, согласно всестороннему сравнительному анализу обучаемых фильтров цветовой гаммы изображений.
4. Разработанная модель обучена с помощью адаптированного и специально реализованного метода регуляризации согласованности, используемого в обучении моделей с частичным привлечением учителя [8, 9], что позволило добиться лучших значений метрик качества.

Также следует отметить, что коррекция цветовой гаммы изображений способствует улучшению качества последующего анализа изображений. Данный эффект достигается путем коррекции недостаточного освещения объекта, избыточного освещения запечатленной сцены, коррекции областей засвета и других артефактов, возникающих при получении цифрового снимка. Для исследования применимости разработанной модели для улучшения качества работы различных классификаторов изображений [10–14], разработанная модель была интегрирована в стек разработанных автором комбинированных нейросетевых классификаторов [10–14]. Данная интеграция позволила улучшить качество классификации изображений различного рода в составе различных комбинированных классификаторов, что описано в данной работе. Таким образом, помимо прямого применения разработанного механизма улучшения цветовой гаммы для улучшения пользовательских фотографий, предлагаемое решение позволяет улучшать качество работы некоторых классификаторов путем предварительной обработки анализируемых изображений.

Постановка задачи

Целью данной работы является создание легковесной нейросетевой модели, содержащей меньшее число параметров, чем сравнимые по эффективности аналоги. При этом, для оценки эффективности улучшения цветовой гаммы фотографий был использован набор данных MIT Adobe FiveK [7]. Этот набор данных содержит 5000 исходных изображений и 5 улучшенных версий (A, B, C, D, E) для каждого из них, предоставленных соответствующим экспертом по ретушированию изображений. Версия C используется в подавляющем большинстве работ по автоматическому улучшению изображений, в качестве эталонного варианта улучшения цветовой гаммы, она выбрана в качестве целевой и при обучении модели, представленной в данной работе. Подмножество RANDOM250 [3,15] набора данных MIT Adobe FiveK использовалось для проверки модели, а обучающее подмножество состояло из оставшихся 4750 пар изображений.

Для расчета расстояния между результатом улучшения и соответствующим целевым изображением, предоставленным экспертом C из набора данных MIT Adobe FiveK, было использовано два общих показателя различия изображений: отношение пикового сигнала к шуму (PSNR) и показатель структурного индекса сходства (SSIM) [16]. Хотя SSIM больше подходит для сравнения локальных артефактов [17], PSNR означает метрику на основе общей среднеквадратичной ошибки (MSE).

Таким образом, задача улучшения цветовой схемы изображения в контексте данной работы формализуется, как создание модели преобразования исходного изображения, минимизирующего расстояние по метрикам PSNR и SSIM между результатом преобразования исходной фотографии из набора RANDOM250 и соответствующей улучшенной версии C. При этом количество параметров созданной модели должно быть минимальным.

Обзор литературы

В настоящее время в области автоматического улучшения цветовой гаммы изображений представлено множество работ с различными подходами к модификации изображения для улучшения визуального восприятия. Методы, основанные на глубоком обучении, на данный момент считаются наиболее успешными в задачах улучшения качества изображений.

Среди широкого разнообразия подходов к улучшению изображений можно выделить работы [2–6, 18–20], в которых предложены методы, демонстрирующие высокое значение метрик качества на наборе данных MIT Adobe FiveK [7].

Примечательный подход к улучшению цветовой гаммы изображений описан в работе [3] и включает применение различных методов ретуширования к входному изображению. Предлагаемое авторами решение, основано на использовании метода глубокого обучения с подкреплением для поиска оптимальной последовательности глобальных преобразований изображения для оптимизации цветовой гаммы. Однако данное решение требует больших вычислительных ресурсов и является очень тяжелым, поскольку оно использует модель VGG-16 [21] и гистограммный анализ для оценки параметров для построения оптимальной стратегии улучшения изображений.

Ну и др. [2] представили аналогичный подход на основе фильтров, который использует интерпретируемые преобразования изображений. Также в данной работе представлен подробный сравнительный анализ и разбор принципов работы фильтров. В предложенной авторами реализации, модель является тяжелой, склонной к неестественным искажениям цвета и обеспечивает низкую скорость работы без реализации дополнительных оптимизаций.

Ди и др. [19] также представили модель коррекции цветовой гаммы изображений, основанную на специально разработанных самоинтерпретируемых фильтрах изображений. Предлагаемая модель эффективно решает проблему отсутствия датасета с попарными соответствиями и устойчива к искажениям, возникающим из-за субъективности экспертов. Из недостат-

ков такого подхода наиболее существенными являются низкие численные значения метрик качества в такой приоритетной составляющей, как коррекция яркости. Например, Pix2Pix [5] показал лучшие результаты. Однако, модель, созданная в рамках данной работы, значительно превосходит Pix2Pix, согласно численным значениям метрик качества.

Другой примечательный подход основан на комбинированной архитектуре, которая разделяет процесс улучшения цветовой гаммы на корректировку по каналам и по пикселям [6]. Это решение использует [22] в качестве кодировщика признаков, нелокальный блок внимания [23] для последующего анализа признаков и глобальное линейное отображение для улучшения характеристик визуального восприятия. *Chai* и др. [20] предложили аналогичный подход, основанный на использовании параметризованного преобразования цвета. Подобные идеи доказали свою эффективность при решении задач визуального улучшения изображения, поэтому комбинированная модель, представленная в данной статье, использует похожие приемы, но при этом и содержит как минимум в 10 раз меньше параметров.

Методы, основанные на генеративно-сопоставительных сетях (GAN) [24], также широко используются для улучшения цветовой гаммы изображения [4, 5]. Обратной стороной этих методов является то, что они имеют склонность порождать артефакты в выходном изображении.

Генератор с глобальными признаками и усовершенствование Wasserstein GAN с адаптивной схемой взвешивания были использованы в двухстороннем методе на основе GAN для задачи улучшения изображений [4]. Однако, несмотря на широкий арсенал прогрессивных приемов, такой подход не демонстрирует конкурентных результатов на наборе данных MIT Adobe FiveK.

EnhanceGAN [18] можно выделить среди других основанных на GAN методов по его конкурентным количественным результатам при решении задач по улучшению цвета на основе метрик эстетики. К сожалению, предложенный авторами кодировщик достаточно тяжелый, что существенно усложняет использование этой архитектуры на мобильных устройствах.

Особого внимания заслуживают методы, основанные на моделях с небольшим количеством параметров. *Moran et al.* [25] предложили легко-

весное решение на основе фильтров, которое учитывает локальные особенности изображений. Однако модель, созданная в рамках настоящей работы, содержит в 8 раз меньше параметров и показывает конкурентоспособные результаты.

Wang и др. [26] представили еще один облегченный движок, который основан на оценке глобальных и локальных характеристик изображения, но описанная ниже модель содержит в 10 раз меньше параметров и также демонстрирует аналогичные результаты на тестовых наборах MIT Adobe FiveK.

Выше рассмотрены несколько современных методов автоматического улучшения изображений, каждый из которых имеет свою концепцию. Подавляющее большинство рассмотренных моделей не подходят для развертывания на мобильных платформах по разным причинам, например из-за большого веса [2–6, 18], вычислительной сложности [2, 3], и низкой скорости работы на реальных данных из-за отсутствия дополнительной оптимизации [2].

Также следует отметить, что предлагаемое в данной работе решение, превосходит большинство известных методов улучшения цветовой гаммы фотографий [25–30] на тестовых наборах MIT Adobe FiveK. Более того, предлагаемая содержит на порядок меньше параметров, чем широко известные легкие механизмы улучшения изображений [20, 25, 26].

В данной работе приведено исследование преобразований изображений, используемых в различных методах улучшения цветовой гаммы изображений. Также представлены результаты специально проведенного сравнительного исследования применения данных преобразований в составе нейросетевой комбинированной схемы. Используя комбинацию исследованных методов, была создана собственная комбинированная модель нейронной сети, которая демонстрирует конкурентоспособные результаты и оптимизирована для использования на мобильных устройствах. Также следует отметить, что разработанная модель (LFIEM [1]), позволяет производить улучшение цветовой гаммы фотографий различной природы (см. Рис. 2, 3).



Рис. 2: Пример автоматического улучшения цветовой гаммы фотографии мебели с помощью модели, разработанной в ходе выполнения данной работы (LFIEM [1]). Оригинальное изображение получено из набора данных Adobe FiveK dataset. (Слева) расположено исходное изображение. (Справа) расположено изображение, полученное с помощью LFIEM [1] из исходного.



Рис. 3: Пример автоматического улучшения цветовой гаммы фотографии ребенка с помощью модели, разработанной в ходе выполнения данной работы (LFIEM [1]). Оригинальное изображение получено из набора данных Adobe FiveK dataset. (Слева) расположено исходное изображение. (Справа) расположено изображение, полученное с помощью LFIEM [1] из исходного.

Глава 1. Архитектура LFIEM

Иллюстрация разработанной комбинированной схемы обработки изображения изображена на Рис. 4. Она состоит из нескольких независимых блоков, количество которых зависит от количества используемых фильтров. В каждом i -м блоке уменьшенная версия исходного изображения I_{so} проходит через генератор параметров h_i , который подбирает параметры p_i для соответствующего фильтра f_i . После применения фильтров к исходному изображению I_o окончательная улучшенная версия изображения I_e представляет собой сумму исходных изображений и выходных результатов после применения специальных фильтров. В общем виде, разработанную модель можно представить следующим образом:

$$I_e = I_o + \sum_{i=1}^n f_i(I_o, h_i(I_{so})),$$

где n - число выбранных фильтров.

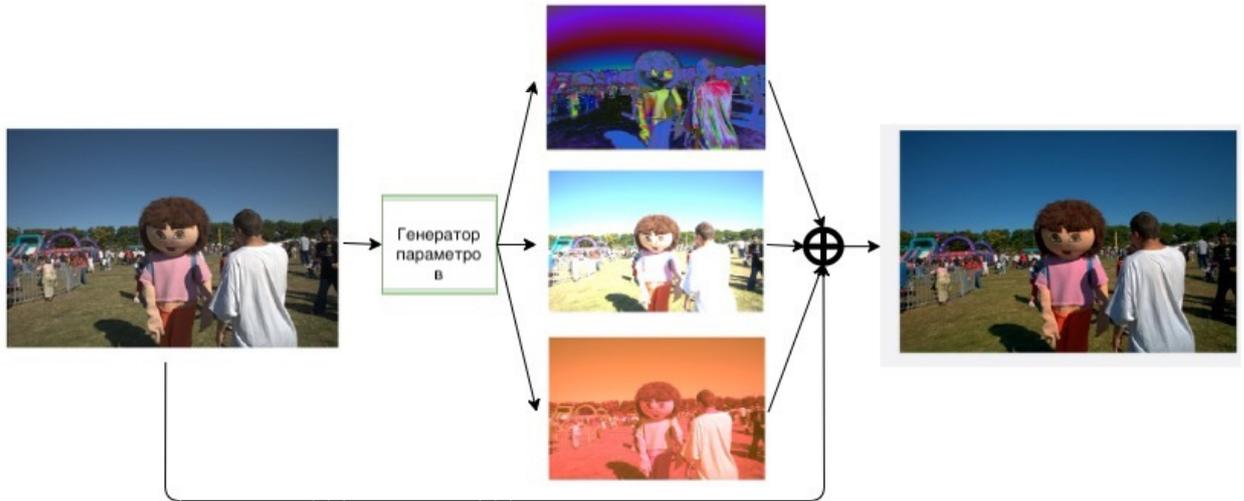


Рис. 4: Эскиз схемы работы LFIEM [1]. Сначала генератор параметров определяет параметры для применения фильтров. Затем все результаты применения фильтров складываются с исходным изображением, после чего получается изображение с улучшенной цветовой гаммой. Исходное изображение для эскиза получено из набора данных MIT Adobe FiveK.

1.1 Генератор параметров

Для облегчения предложенной архитектуры, генератор параметров содержит всего две стадии, реализованные в виде отдельных конструктивных элементов (см. Рис. 5). Первая стадия реализована с помощью трех сверточных слоев с шагом, равным двум точечным позициям на промежуточной карте признаков. За каждым сверточным слоем, кроме первого, следует слой нормализации [31] и активация LeakyReLU [32]. При этом, количество карт признаков установлено равным 16, 32 и 128 в первом, втором и третьем сверточном слое соответственно. Второй этап состоит из двух полносвязных слоев с активацией ReLU между ними. Наличие функции результирующей активации зависит от того, за какой фильтр отвечает генератор текущего параметра. В предлагаемой архитектуре (LFIEM [1]) используется сигмоидальная функция для диапазона $[0, 1]$, функция \tanh для $[-1, 1]$ и тождественная функция активации используется, если диапазон не был ограничен для фильтра.

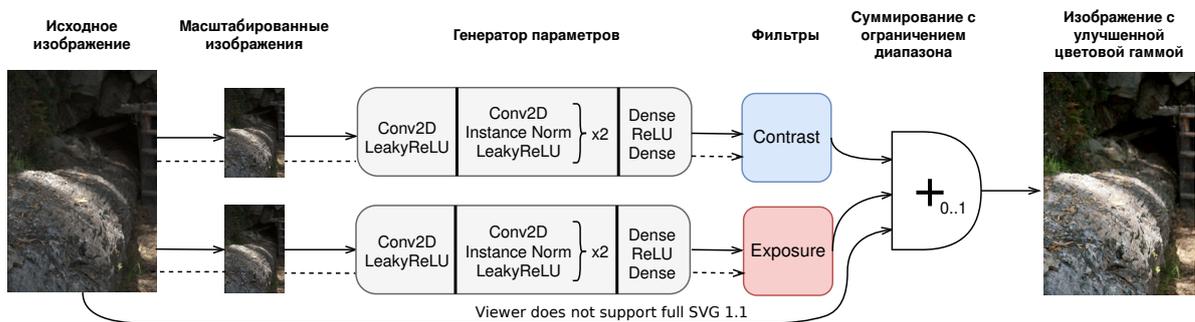


Рис. 5: Генератор параметров в составе LFIEM [1], сконфигурированной фильтрами контрастности и экспозиции. .

1.2 Преобразования изображений

Для построения модели LFIEM [1] был выбран ряд дифференцируемых преобразований изображения (фильтров). Примеры используемых фильтров показаны на Рис. 6.



(1) Оригинальное



(2) Насыщение



(3) Контраст



(4) Баланс белого



(5) Экспозиция



(6) Универсальное ядро



(7) Резкость



(8) Линейное преобразование



(9) Поканальное преобразование

Рис. 6: Оригинальное изображение из набора данных MIT Adobe FiveK и результаты применения 8 исследуемых фильтров, инициализированных случайными параметрами.

В данном разделе используется унифицированное обозначение фильтров: I_{in} - входное изображение, I_{out} - выходное изображение, (x, y) - координаты пикселей изображения, c обозначает цветовой канал (красный, зеленый или синий), а p, q, r, s, t, u являются обучаемыми параметрами, если не указано иное.

Чтобы добиться хорошей производительности по времени, используется только цветовое пространство RGB для представления изображения. Для единообразного описания используются значения каналов в диапазоне $[0, 1]$.

В рамках данной работы была рассмотрена автоматическая коррекция насыщенности цвета, которая реализуется с помощью преобразования, описанного выше, где $p \in [-1, 1]$, представляет собой обучаемый параметр, характеризующий преобразование применяемое к каждому пикселю входного изображения. В общем виде, отображение можно описать следующей формулой:

$$\Delta[x, y] = \begin{cases} (m - I_{in}[x, y]) \cdot (1 - \frac{1}{1-p}), & \text{если } p > 0 \\ -(m - I_{in}[x, y]) \cdot p, & \text{иначе;} \end{cases}$$

$$I_{out}[x, y] = I_{in}[x, y] + \Delta[x, y],$$

где m обозначает среднее значение пикселей входного изображения по каналам.

Помимо насыщенности цвета, была рассмотрена автоматическая коррекция контрастности, которая задается представленным ниже преобразованием исходного изображения, где $p \in [-1, 1]$, представляет собой единственный обучаемый параметр, который определяет преобразование, применяемое к каждому пикселю входного изображения.

В общем виде, преобразование, применяемое к исходному изображению с целью коррекции контрастности, выглядит следующим образом:

$$I_{out}[x, y] = \begin{cases} (I_{in}[x, y] - 0.5) \cdot \frac{1}{1-r}, & \text{если } r > 0 \\ (I_{in}[x, y] - 0.5) \cdot (1 - r), & \text{иначе;} \end{cases}$$

Автокоррекция баланса белого выражена в терминах умножения значения цветового канала каждого пикселя входного изображения на обучаемый параметр в диапазоне $[0, 1]$:

$$I_{out}^c[x, y] = I_{in}[x, y] \cdot s_c.$$

Для данного преобразования использовалось три обучаемых параметра для красного, зеленого и синего каналов соответственно.

Следующее преобразование изображений позволяет эмулировать автоматическую коррекцию экспозиции:

$$I_{out}[x, y] = I_{in}[x, y] \cdot 2^t.$$

Также были исследованы фильтры на основе свертки со специальным ядром. Наиболее примечательные результаты преобразований такого типа были получены с использованием комбинированного ядра с маской и полностью обучаемой сверткой. Для краткости, назовем такой тип ядра универсальным ядром. Варьируя маску, возможно модифицировать преобразование на основе универсального ядра. Примером этому может являться использование специального ядра, добавляющего резкость исходному изображению.

Классическое представление сверточного фильтра универсальным ядром выглядит следующим образом:

$$I_{out} = I_{in} \circledast \frac{1}{\nu} P,$$

где $P \in \mathbb{R}^{3 \times 3}$ - полностью обучаемая матрица ядра фильтра, а ν - сумма элементов P для нормализации матрицы.

Фильтр резкости определяется с помощью вспомогательной формулы:

$$I_{out} = I_{in} \circledast \frac{1}{\nu} (K + M \cdot q),$$

где K - матрица ядра фильтра, M - матрица отображения той же формы, что и K , а ν - сумма элементов $(K + M \cdot q)$ для нормализации матрицы ядра. Вышеупомянутая формула применяется к красному, зеленому и

синему каналам независимо с собственным обучаемым параметром. Таким образом, модификация резкости определяется следующими параметрами:

$$K = \begin{pmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & -476 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix},$$

$$M = \begin{pmatrix} 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \\ 0.8 & 0.9 & 0.9 & 0.9 & 0.8 \\ 0.8 & 0.9 & 1 & 0.9 & 0.8 \\ 0.8 & 0.9 & 0.9 & 0.9 & 0.8 \\ 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \end{pmatrix},$$

Обучаемое линейное преобразование изображения, описанное в [6], является еще одним примечательным фильтром. Данное преобразование можно описать следующим выражением:

$$I_{out}[x, y] = P \cdot I_{in}[x, y] + b,$$

где $P \in \mathbb{R}^{3 \times 3}$ обозначает обучаемую матрицу аффинного преобразования, $b \in \mathbb{R}^3$ - обучаемый вектор в цветовом пространстве RGB.

Также использовалось отображение, описывающее поканальное преобразование цвета, описанное в [33]. Данное преобразование состоит из тройки функций, которые применяются к красному, зеленому и синему цветовым каналам соответственно. Каждая функция представляет собой линейную комбинацию элементов f_1, f_2, \dots, f_n n -мерного базиса, коэффициенты для которых вычисляются из выходных данных нейронной сети. Следовательно, значение канала для каждого пикселя входного изображения вычисляется по следующей формуле:

$$I_{out}^c[x, y] = I_{in}^c[x, y] + \sum_{i=1}^n u_{ic} \cdot f_i(I_{in}^c[x, y]),$$

где f_1, f_2, \dots, f_n - упомянутый выше функциональный базис, а u_c - обучаемые параметры (по одному параметру на каждый канал).

Из-за доказанной эффективности [33] был рассмотрен только кусочный базис и использован следующий набор функций:

$$f_i(x) = \max(0, 1 - |(n - 1) \cdot x - i + 1|), i \in \{1, 2, \dots, n\},$$

где x - значение текущего пикселя входного изображения.

1.3 Функция потерь с регуляризацией согласованности

В данной работе была произведена адаптация регуляризации согласованности, представляющей собой хорошо известный метод обучения с частичным привлечением учителя, для задачи улучшения цветовой гаммы изображения. Такой подход не влияет на размер или время работы модели и лишь незначительно усложняет процесс обучения.

Генераторы параметров в LFIEM [1] должны давать одинаковые результаты для похожих изображений, что означает, что выходные данные генератора должны быть инвариантными относительно любых слабых изменений изображений.

По этой причине регуляризация согласованности была использована во время обучения. В терминах обозначений из начала этого раздела, регуляризация согласованности L_{cr} определяется следующим образом:

$$L_{cr} = \lambda \sum_{i=1}^n \|h_i(I_{so}) - h_i(T(I_{so}))\|^2,$$

где λ - весовой коэффициент, $T(\cdot)$ - случайная функция незначительной модификации изображения, а $\|\cdot\|$ обозначает норму L^2 данного вектора.

В данной работе применялась комбинация случайного сдвига изображения на несколько пикселей, а затем случайного переворачивания изображения по горизонтали и вертикали в качестве незначительной модификации изображения.

В первую очередь в качестве функций потерь были использованы L_1 и L_{SSIM} , которые задаются следующим образом:

$$\begin{aligned} L_1 &= \|I_e - I_{gt}\|_1, \\ L_{SSIM} &= 1 - SSIM(I_e, I_{gt}), \end{aligned}$$

где $\|\cdot\|_1$ - это L^1 - норма данного вектора, SSIM - это значение индекса структурного сходства [16], I_e и I_{gt} обозначают улучшенное изображение и целевое изображение соответственно.

Таким образом, модель была обучена с комбинацией L_1 , L_{SSIM} и L_{cr} :

$$L = L_1 + L_{SSIM} + L_{cr}.$$

Глава 2. Постановка экспериментов

2.1 Набор данных

Для оценки модели был использован набор данных MIT Adobe FiveK [7]. Этот набор данных содержит 5000 исходных изображений и 5 улучшенных версий (A, B, C, D, E) для каждого из них, предоставленных соответствующим экспертом по ретушированию изображений. Версия C использовалась в подавляющем большинстве работ по автоматическому улучшению изображений, поэтому и была выбрана в качестве целевой для разработанной модели LFIEM [1]. Подмножество RANDOM250 [3, 15] набора данных MIT Adobe FiveK использовалось для проверки модели, а обучающее подмножество состояло из оставшихся 4750 пар изображений.

Также были использованы и другие конфигурации расщепления исходного набора данных, для того, чтобы воспроизвести результаты, полученные в других работах. Следует отметить, что для обучения, проверки и сравнения моделей было использовано только цветовое пространство RGB.

2.2 Метрики качества

Для расчета численных отличий расстояния между результатом улучшения цветовой гаммы и соответствующим целевым изображением, предоставленным экспертом С из набора данных MIT Adobe FiveK, было использовано два общих показателя различия изображений: отношение пикового сигнала к шуму (PSNR) и показатель индекса структурного сходства (SSIM) [16]. SSIM стремится сравнить структурную информацию из взаимозависимости значений пикселей, что позволяет учитывать структурные особенности сцены и представленных объектов. В то время, как SSIM больше подходит для сравнения локальных артефактов [17], PSNR представляет собой метрику на основе общей среднеквадратичной ошибки).

2.3 Детали реализации

Предложенная модель LFIEM [1] была построена с помощью TensorFlow 2 [34] и оптимизирована для работы на мобильных устройствах с помощью TensorFlow Lite. Вся предварительная обработка данных выполнялась аналогично [6]: изображения были дополнены до размера 500×500 , а затем нормализованы до $[0, 1]$. Модель была обучена с помощью оптимизатора Adam со следующими значениями гиперпараметров: $\beta_1 = 0$, $\beta_2 = 0.9$ и размером пакета 40. Скорость обучения была инициализирована равной $1e^{-3}$ и уменьшалась в 0,95 раза на каждые 1200 шагов. Для всех экспериментов обучение прекращалось после 10000 шагов.

Глава 3. Анализ полученных результатов

В этом разделе описана проверка разработканной модели (LFIEМ [1]) с помощью трех экспериментов. Во-первых, были проанализированы различные комбинации фильтров и определены наиболее эффективные. Во-вторых, продемонстрировано преимущества использования регуляризации согласованности для наиболее успешных наборов фильтров. Наконец, произведено сравнение LFIEМ [1] с моделями из других работ.

В этом разделе использованы следующие сокращения для обозначения нескольких фильтров: `c-w color` для цвета по каналам, `u-kernel` для универсального ядра и `w-balance` для баланса белого.

3.1 Анализ фильтров

Чтобы определить, какие фильтры больше всего влияют на улучшение цветовой гаммы изображений, было проведено сравнительное исследование. На основе предварительно выбранных фильтров из раздела 1.2 был сгенерирован широкий спектр комбинаций из одного, двух и трех фильтров и поведена оценка каждой полученной конфигурации с помощью метрики SSIM. Каждая конфигурация была обучена пять раз с разными случайными инициализациями, после чего посчитаны средние значения и дисперсия для метрики SSIM. Результаты для комбинаций фильтров, которые показали себя лучше всего, можно увидеть в таблице 1.

Основываясь на результатах проведенного исследования, можно сделать вывод, что поканальные цветные и линейные фильтры оказывают наибольшее влияние на качество улучшения цветовой гаммы изображений. Вопреки изначальным ожиданиям, свертка с универсальным ядром не показала лучших результатов в связке с другими преобразованиями.

Таблица 1: Сравнительный анализ различных конфигураций фильтров на наборе данных RANDOM250. В таблице представлены только лучшие 15 результатов.

Комбинация фильтров	SSIM
saturation, sharp, c-w color	0.9049 (± 0.0013)
exposure, contrast, c-w color	0.9056 (± 0.0008)
c-w color	0.9058 (± 0.0017)
contrast, c-w color	0.9059 (± 0.0007)
saturation, u-kernel, c-w color	0.9062 (± 0.0012)
exposure, saturation, c-w color	0.9064 (± 0.0017)
w-balance, c-w color	0.9066 (± 0.0007)
saturation, c-w color	0.9071 (± 0.0005)
u-kernel, linear, c-w color	0.9076 (± 0.0012)
saturation, c-w color, w-balance	0.9081 (± 0.0010)
saturation, contrast, c-w color	0.9081 (± 0.0008)
linear, c-w color	0.9082 (± 0.0005)
linear, c-w color, w-balance	0.9083 (± 0.0012)
contrast, linear, c-w color	0.9086 (± 0.0015)
exposure, linear, c-w color	0.9087 (± 0.0008)

3.2 Влияние регуляризации согласованности

Регуляризация согласованности (Consistency Regularization, CR) - это подход, который активно используется в обучении с частичным привлечением учителя. Основная идея этого метода относительно применения в области компьютерного зрения - обеспечить согласованный вывод модели в случае слабой дисперсии между входными изображениями. При обучении LFIEM [1] была использована та же идея (случайное слабое изменение изображения не должно влиять на параметры обучаемого фильтра). Под случайным слабым изменением изображения мы понимаем случайный сдвиг на несколько пикселей (0-4) и случайное горизонтальное и вертикальное переворачивание изображения.

Экспериментальные результаты, показанные в таблице 2, демонстрируют важность применения регуляризации согласованности для решаемой задачи. Полученные результаты также свидетельствуют о том, что использование регуляризации согласованности, особенно для комбинации фильтров контраста, линейного преобразования, и поканального цветового преобразования, приводит к значительному приросту значений метрик качества по сравнению с экземплярами модели, использующими оптимальные конфигурации фильтров, но в при обучении которых она не использовалась.

Таблица 2: Наиболее успешные конфигурации фильтров с применением регуляризации согласованности. (Значения метрик получены на тестовом наборе RANDOM250).

Комбинация фильтров	CR	SSIM
linear, c-w color	55	0.9082 (± 0.0005)
linear, c-w color	51	0.9092 (± 0.0008)
linear, c-w color, w-balance	55	0.9083 (± 0.0012)
linear, c-w color, w-balance	51	0.9091 (± 0.0010)
contrast, linear, c-w color	55	0.9086 (± 0.0015)
contrast, linear, c-w color	51	0.9103 (± 0.0005)
exposure, linear, c-w color	55	0.9087 (± 0.0008)
exposure, linear, c-w color	51	0.9093 (± 0.0004)

3.3 Сравнительный анализ с другими методами

В ходе данной работы также было проведено сравнение LFIEM [1], представленной в лучшей конфигурации с современными методами, описанными в разделе . Результаты сравнения представлены в таблице 3.

Таблица 3: Сравнение различных методов коррекции цветовой гаммы изображений и лучшей конфигурации LFIEM [1] на наборе данных MIT Adobe FiveK dataset (в цветовом пространстве RGB).

Метод	# параметров	PSNR	SSIM	Train-test разбиение
Pix2Pix [5]	54M	-	0.857	4750-250
Distort-and-Recover [3]	153M	-	0.905	4750-250
DPED [27]	-	21.76	0.871	2250-500
8RESBLK [28]	-	23.42	0.875	2250-500
CRN [35]	-	22.38	0.877	2250-500
HDRNet [29]	-	21.96	0.866	4500-500
SULPCE [20]	>1M	23.93	0.920	4000-1000
U-Net [30]	1.3M	22.24	0.850	4500-500
DeepUPE [26]	1.0M	23.04	0.893	4500-500
DeepLPF [25]	800K	24.48	0.887	4500-500
DPE [4]	2.2M	23.89	0.906	4750-250
CE+PRNL [6]	>30M	24.19	0.915	4750-250
LFIEM [1]	101K	24.77	0.911	4750-250

Несмотря на то, LFIEM [1] направлена на адаптацию для использования на мобильных устройствах, данная модель показывает конкурентоспособные результаты по сравнению с намного более тяжелыми моделями, созданными с целью получения лишь самых высоких численных значений метрик качества. Еще одно преимущество LFIEM [1] - отсутствие локальных артефактов и глобального искажения цветового пространства. Последнюю проблему можно увидеть в Exposure [2] (см. Рисунок 7).



Рис. 7: Примеры результатов применения методов улучшения цветовой гаммы изображений. В каждой паре, слева представлен результат применения метода Exposure [2] к оригинальному изображению, а справа - метода LFIEM [1]. Оригинальные изображения выбраны из набора данных MIT Adobe FiveK dataset [7].

Глава 4. Интеграции LFIEM в комбинированные архитектуры классификации изображений

Согласно сказанному ранее, отдельный интерес представляет возможность применения разработанной нейросетевой модели для улучшения цветовой гаммы изображений (LFIEM) на стадии предобработки при решении задачи классификации изображений в сложных условиях. Данный интерес обусловлен способностью LFIEM корректировать влияние внешних факторов в процессе получения снимков.

4.1 Задача классификации изображений в сложных условиях

В общем виде, задача классификации изображений, может быть сформулирована следующим образом. Исходной изображение Q должно быть отнесено к одному из классов $C = C_i$, где $i \in [0, N]$.

Отметим, что помимо формальной постановки задачи классификации, к решению, работающему в сложных условиях, предъявляются требования по устойчивости к различным условиям съемки. В частности, допускаются засветы, затенения и другие артефакты, возникающие из-за варьирования освещения и ракурса съемки.

Таким образом, исходные изображения получены с помощью камер различного типа при разных условиях, следовательно:

- а) могут содержать визуальные дефекты – засветы и затенения;
- б) цветовой баланс изображений может быть нарушен.

4.2 Интеграция LFIEM в комбинированный классификатор удостоверений личности

Для анализа возможности применения LFIEM в качестве модуля предварительной обработки в составе комбинированных нейросетевых классификаторов изображений, LFIEM была интегрирована в классификатор удостоверений личности [13], разработанный автором данной работы в рамках исследований комбинированных нейросетевых архитектур. Данная комбинированная схема производит классификацию ограниченных фрагментов фотографий удостоверений личности в сложных условиях освещения при различных ракурсах съемки [13]. Для уменьшения негативного влияния недостатка или избытка освещения, в состав исходного классификатора была интегрирована LFIEM, производящая предобработку исходных изображений таким образом, что на вход исходного классификатора подается не исходное изображение, а результат его обработки с помощью LFIEM. Метрики качества работы исходного классификатора и модифицированного с помощью внедрения стадии предобработки изображений с помощью LFIEM, представлены в таблице 4.

Конфигурация классификатора	F_1
VGG19 [21]	0.66
ResNet [22]	0.71
AE+VGG19 [13, 21]	0.77
AE+ResNet [13, 22]	0.82
VCA [1, 13]	0.91
LFIEM + VGG19 [1, 21]	0.68
LFIEM + ResNet [1, 22]	0.75
LFIEM + AE+VGG19 [1, 13, 21]	0.8
LFIEM + AE+ResNet [1, 13, 22]	0.87
LFIEM + VCA [1, 13]	0.96

Таблица 4: Значения F_1 на тестовых выборках для различных конфигураций комбинированного классификатора удостоверений личности.

4.3 Интеграция LFIEM в комбинированный классификатор фотографий фасадов коммерческих зданий

Также в ходе исследования возможности применения LFIEM в качестве модуля предварительной обработки в составе комбинированных нейросетевых классификаторов изображений, LFIEM была интегрирована в комбинированный классификатор фасадов фотографий коммерческих зданий по типу предоставляемых услуг, разработанный автором данной работы [10, 12, 14]. Следует также отметить, что рассматриваемый комбинированный классификатор также работает в сложных условиях освещения, так как съемка производится в условиях варьирующегося уличного освещения [10–12, 14]. Для уменьшения негативного влияния недостатка или избытка освещения, в исходный классификатор также была интегрирована LFIEM на стадии предварительной обработки. Метрики качества работы исходной модели и улучшенной с помощью LFIEM представлены в таблице 5.

Конфигурация классификатора	F_1
Type A [12]	0.59
Type B [12]	0.63
LFIEM + Type A [1, 12]	0.62
LFIEM + Type B [1, 12]	0.68

Таблица 5: Значения F_1 на тестовых выборках для различных конфигураций комбинированного классификатора фасадов коммерческих зданий.

4.4 Анализ результатов интеграции

Исходя из приведенных выше численных значений метрик качества, следует, что использование предобработки в виде применения LFIEM к исходным изображениям, позволяет увеличить эффективность работы рассмотренных комбинированных нейросетевых классификаторов, путем уменьшения негативного влияния сложных внешних условий на цветовую гамму изображений, что позволяет повысить качество последующего анализа изображений.

Заключение

В ходе данной работы была создана облегченная модель улучшения изображений (LFIEM [1]), преимуществами которой является интерпретируемая архитектура и небольшое количество весов (101547 обучаемых параметров) позволяет разворачивать полученное решение на мобильных устройствах. Кроме того, LFIEM [1] превосходит подавляющее большинство известных моделей улучшения цветовой гаммы изображений и содержит как минимум в десять раз меньше параметров, чем большинство из них. Также для обучения LFIEM [1] был адаптирован современный подход регуляризации согласованности, который позволил достичь лучших результатов по сравнению с другими моделями. Также был проведен анализ применимости и интеграция разработанной модели в стеки различных комбинированных классификаторов, разработанных автором [10–14], в результате чего был обнаружен прирост качества классификации, что продемонстрировало эффективность применимости данной нейросетевой модели на стадии предобработки для последующей классификации изображений. В процессе разработки LFIEM [1] был проведен всесторонний сравнительный анализ различных преобразований, используемых для улучшения цветовой гаммы изображений. Результаты данной работы были представлены на ряде международных и российских конференций: ICPR 2020 (25th International Conference on Pattern Recognition), Baltic DBIS 2020 (14th International Baltic Conference on Databases and Information Systems) - Best Short Paper Award, AIST 2020 (10th International Conference on Analysis of Images, Social Networks and Texts), AIST 2019 (9th International Conference on Analysis of Images, Social Networks and Texts), SEIM 2020 (5th Conference on Software Engineering and Information Management) и опубликованы в соответствующих сборниках и журналах [1, 10–14].

Список литературы

- [1] O. Tatanov and A. Samarin, “Lfiem: Lightweight filter-based image enhancement model,” in 2020 25th International Conference on Pattern Recognition (ICPR), 2021, pp. 873–878.
- [2] Y. Hu, H. He, C. Xu, B. Wang, and S. Lin, “Exposure: A white-box photo post-processing framework,” *ACM Transactions on Graphics (TOG)*, vol. 37, no. 2, p. 26, 2018.
- [3] J. Park, J.-Y. Lee, D. Yoo, and I. So Kweon, “Distort-and-recover: Color enhancement using deep reinforcement learning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5928–5936.
- [4] Y.-S. Chen, Y.-C. Wang, M.-H. Kao, and Y.-Y. Chuang, “Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6306–6314.
- [5] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.
- [6] C. Shan, Z. Zhang, and Z. Chen, “A coarse-to-fine framework for learned color enhancement with non-local attention,” in *2019 IEEE International Conference on Image Processing (ICIP)*, Sep. 2019, pp. 949–953.
- [7] V. Bychkovsky, S. Paris, E. Chan, and F. Durand, “Learning photographic global tonal adjustment with a database of input/output image pairs,” in *CVPR 2011. IEEE*, 2011, pp. 97–104.
- [8] Y. Reddy, V. Pulabaigari, and E. B, “Semi-supervised learning: a brief review,” *International Journal of Engineering Technology*, vol. 7, p. 81, 02 2018.
- [9] M. Sajjadi, M. Javanmardi, and T. Tasdizen, “Regularization with stochastic transformations and perturbations for deep semi-supervised learning,” in *Advances in Neural Information Processing Systems*, 2016, pp. 1163–1171.
- [10] A. Samarin and V. Malykh, *Ensemble-Based Commercial Buildings Facades Photographs Classifier. Analysis of Images, Social Networks and Texts (AIST)*, Springer International Publishing, 04 2021, pp. 257–265.
- [11] —, Worm-like image descriptor for signboard classification. *Proceedings of the Fifth Conference on Software Engineering and Information Management 2020 (SEIM 2020)*, CEUR Workshop Proceedings, 05 2020.

- [12] A. Samarin, V. Malykh, and S. Muravyov, Specialized Image Descriptors for Signboard Photographs Classification. *Databases and Information Systems*, Springer International Publishing, 08 2020, pp. 122–129.
- [13] А.В. Самарин, В.А. Малых, Метод верификации изображений удостоверений личности по ограниченному фрагменту изображения. *Труды Института системного анализа РАН*, 2020, pp. 15–23.
- [14] V. Malykh and A. Samarin, “Combined advertising sign classifier,” in *Analysis of Images, Social Networks and Texts*. Cham: Analysis of Images, Social Networks and Texts (AIST), Springer International Publishing, 2019, pp. 179–185.
- [15] Z. Yan, H. Zhang, B. Wang, S. Paris, and Y. Yu, “Automatic photo adjustment using deep neural networks,” *ACM Transactions on Graphics (TOG)*, vol. 35, no. 2, p. 11, 2016.
- [16] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli et al., “Image quality assessment: from error visibility to structural similarity,” *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [17] A. Hore and D. Ziou, “Image quality metrics: Psnr vs. ssim,” in *2010 20th International Conference on Pattern Recognition*. IEEE, 2010, pp. 2366–2369.
- [18] Y. Deng, C. C. Loy, and X. Tang, “Aesthetic-driven image enhancement by adversarial learning,” *10 2018*, pp. 870–878.
- [19] X. Du, X. Yang, Z. Qin, and J. Tang, “Progressive image enhancement under aesthetic guidance,” *06 2019*, pp. 349–353.
- [20] Y. Chai, R. Giryes, and L. Wolf, “Supervised and unsupervised learning of parameterized color enhancement,” *03 2020*, pp. 981–989.
- [21] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *CoRR*, vol. abs/1409.1556, 2014.
- [22] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [23] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7794–7803.
- [24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

- [25] S. Moran, P. Marza, S. McDonagh, S. Parisot, and G. Slabaugh, “Deeplpf: Deep local parametric filters for image enhancement,” 03 2020.
- [26] R. Wang, Q. Zhang, C.-W. Fu, X. Shen, W.-S. Zheng, and J. Jia, “Underexposed photo enhancement using deep illumination estimation,” 06 2019, pp. 6842–6850.
- [27] A. Ignatov, N. Kobyshev, R. Timofte, K. Vanhoey, and L. Van Gool, “Dslr-quality photos on mobile devices with deep convolutional networks,” 10 2017, pp. 3297–3305.
- [28] J. Chen, A. Adams, N. Wadhwa, and S. Hasinoff, “Bilateral guided upsampling,” *ACM Transactions on Graphics*, vol. 35, pp. 1–8, 11 2016.
- [29] M. Gharbi, J. Chen, J. Barron, S. Hasinoff, and F. Durand, “Deep bilateral learning for real-time image enhancement,” *ACM Transactions on Graphics*, vol. 36, 07 2017.
- [30] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” 05 2015.
- [31] D. Ulyanov, A. Vedaldi, and V. Lempitsky, “Instance normalization: The missing ingredient for fast stylization,” *arXiv preprint arXiv:1607.08022*, 2016.
- [32] A. L. Maas, A. Y. Hannun, and A. Y. Ng, “Rectifier nonlinearities improve neural network acoustic models,” in *ICML*, 2013.
- [33] S. Bianco, C. Cusano, F. Piccoli, and R. Schettini, “Learning parametric functions for color image enhancement,” in *International Workshop on Computational Color Imaging*. Springer, 2019, pp. 209–220.
- [34] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard et al., “Tensorflow: A system for large-scale machine learning,” in *OSDI*, 2016, pp. 265–283.
- [35] Q. Chen and V. Koltun, “Photographic image synthesis with cascaded refinement networks,” 10 2017, pp. 1520–1529.