

PyEmbroider

Давидюк Тимофей, Иванов Никита, Ковырзин Павел

Проект представляет собой телеграм-бота, который по отправленной ему фотографии генерирует схему для вышивания крестиком, учитывая те цвета ниток, которые есть у пользователя, а также предлагая купить недостающие.

Пользовательская документация

1. Найти бота можно по имени @PyEmbroider_bot
2. По командам /start и /help можно получить приветственное сообщение и небольшое сообщение о возможностях бота.
3. Имеются 4 основные кнопки: 📁 Загрузить изображение, 📁 Посмотреть мои фото, 🌈 Добавить цвета, ✅ Показать добавленные цвета.
4. По кнопке «📁 Загрузить изображение» пользователь лишь получает сообщение «Отправьте мне изображение, которое вы хотите преобразовать в схему». Конечно, отправить изображение боту можно и предварительно не нажимая на эту кнопку.
5. По кнопке «📁 Посмотреть мои фото» пользователь получает список сохраненных изображений. Имеется возможность посмотреть каждое изображение, нажав на кнопку с названием файла, а также удалить изображение, нажав на кнопку «❌ удалить «имя изображения»». Бот возвращает именно фото, которое загружал пользователь, а не схемы вышивки по следующим причинам: во-первых пользователь сможет повторно использовать фото для создания новой схемы (например, если у него изменился набор ниток), а во-вторых файл с одним лишь изображением занимает меньше места в памяти.
6. По кнопке «🌈 Добавить цвета» пользователь получает сообщение «Выберите бренд ниток:» и 3 бренда на выбор: «DMC», «Anchor», «Cosmo». Выбрав какой-нибудь бренд (например, «DMC»), пользователь получает сообщение «Выберите группу цветов для бренда DMC:» и кнопки для выбора цвета («Белые», «Красные», «Розовые» и т.д.). При выборе цвета пользователь получает фото с доступными оттенками данного цвета, их названиями, а также кодами. Далее, нажимая на кнопки доступных оттенков данного цвета пользователь может добавлять их в свои цвета (при первом нажатии), а также удалять (при втором нажатии). При этом он получает сообщения вида «Цвет '902' добавлен».
7. При нажатии на кнопку «✅ Показать добавленные цвета» пользователь получает 3 сообщения. Первое сообщение соответствует добавленным ниткам бренда «DMC», второе сообщение соответствует добавленным ниткам бренда «Anchor», третье - ниткам «Cosmo». Если среди добавленных ниток есть хотя бы один оттенок данного бренда, пользователь получает изображение с выбранными оттенками данного бренда и их названиями. Иначе он получает сообщение вида «Пока что не добавлено ни одного цвета бренда Anchor».
8. Процесс загрузки фото. Пользователь может загрузить изображение в любой момент использования бота. Изображение автоматически сохраняется и будет показано при нажатии на кнопку «📁 Посмотреть мои фото». Если пользователь хочет задать изображению имя, он вводит его в качестве подписи к фото, иначе в качестве имени изображению присваивается уникальный uuid шифр. Бот не принимает иные виды файлов: видео, GIF и т.д.
9. Процесс создания схемы. При загрузке изображения пользователь в ответ получает: 1) сообщение вида «Изображение сохранено под именем: dbe27a8f-4f26-4b94-bc36-9bb744e2798a.jpg», 2) «Схема на основе всех возможных цветов:» и прикрепленный pdf файл, 3) «Схема на основе Ваших цветов:» и прикрепленный pdf файл, если пользователь добавил хотя бы один цвет, и сообщение «Сформировать схему на основе ваших цветов невозможно. Вы не добавили ни одного цвета.» иначе. Первый pdf файл содержит пикселизацию изображения. Каждый пиксель имеет один из цветов ниток, которые есть в базе бота. Также файл имеет легенду, где вынесены все цвета, которые есть в пикселизации, написано их название, код и бренд. Аналогичную схему содержит второй pdf файл, за тем исключением, что в пикселизации используются лишь цвета, добавленные пользователем.

Алгоритм пикселизации

Алгоритм пикселизации изображения, реализованный в представленной программе, включает несколько ключевых шагов, направленных на преобразование обычного изображения в его пикселизованную версию с использованием заданной цветовой палитры и техник замены цветов.

Для начала программа импортирует необходимые библиотеки: PIL, предназначенную для работы с изображениями, и Counter из модуля collections, которая используется для подсчета уникальных цветов в изображении. Затем в функции `closest_color` реализуется поиск ближайшего цвета из доступных, принимая на вход заданный цвет и список доступных цветов. Это достигается с помощью метрики Евклида, которая позволяет вычислить квадратное расстояние в RGB-пространстве между цветами.

Следующим важным шагом является функция `replace_rare_colors`, в которой происходит замена редко встречающихся цветов на более распространенные. Функция анализирует все пиксели изображения и определяет, какие из них встречаются реже заданного порога, именуемого `rarity_threshold`. Для каждого редкого цвета находится ближайший часто встречающийся цвет, который будет использоваться для замены, после чего пиксели изображения обновляются на основе этого нового цветового отображения.

Основная функция `pixelate` отвечает за саму пикселизацию изображения. Сначала изображение масштабируется, если его минимальный размер меньше 700 пикселей, что улучшает качество пикселизации. Затем изображение уменьшается в размерах, создавая эффект пикселизации, при этом применяется правило, зависящее от минимального размера. Каждому пикселю присваивается ближайший доступный цвет, после чего вызывается функция `replace_rare_colors` для улучшения цветовой палитры изображения. В финале изображение снова увеличивается до своих исходных размеров.

Для создания акцента на пикселях в изображении добавляются черные границы. Это достигается путем выполнения двух проходов по изображению: первый проход для рисования границ между обычными пикселями, а второй — для рисования более выраженных границ каждые 10 пикселей. Толщина границ определяется параметром `margin_thickness`.

Планируемые улучшения

1) На данный момент изображения пользователей хранятся локально в директории компьютера, а информация о введенных пользователем цветах - в словаре. Для хранения этих данных будет организована база данных.

2) Пока что бот умеет делать лишь пикселизацию изображения. Это неудобно для восприятия, т.к. некоторые пиксели могут иметь похожие оттенки и быть трудно различимыми. Чтобы это исправить, планируется добавить в pdf файлы еще одно изображение - схему вышивки, где цветные пиксели будут заменены, например, числами (пиксели одного цвета - одними числами, пиксели разных цветов - разными).

3*) Еще одна проблема заключается в том, что получаемые схемы не всегда удобны для вышивания. А именно, возможна ситуация, когда встречается много довольно редких пикселей каких-то оттенков. Чтобы этого избежать, предлагается заменять цвет этих пикселей на цвет пикселей, их окружающих. Мы попробуем решить эту задачу, но ничего не обещаем :)