# 4 Предлагаемое научно-техническое решение

В своей статье автор утверждает что пороговое значение Threshold используемое при погружения битов ЦВЗ было подобрано экспериментально, при этом конкретные значения Threshold для каждого изображения в статье не приводится. Автором лишь сказано что параметр подбирался для каждого пары изображений (покрывающего и погружаемого) отдельно, так чтобы параметры качества удовлетворяли следующим значениям:

Таблица

|  |  |
| --- | --- |
| Параметр качества | Значение |
| NC | > 0.8 |
| SSIM | >0.99 |
| PSNR | >45 |

Предлагаемое техническое решение содержит ряд улучшений рассмотренного алгоритма DDFA. В основе решения лежит методика кооперационной гибридной оптимизации, где два разных алгоритма выполняют две разные задачи:

\* Поиск оптимального значения Threshold будет осуществляться с использованиям алгоритма Светлячков.

\* Выбора оптимальных блоков для встраивания ЦВЗ будет осуществляться при помощи Генетического алгоритма.

## 4.1 Описание этапов работы алгоритма

### 4.1.1 Этап 0. Предварительная обработка изображения

На первом этапе файл с черно-белым изображением размером 512х512 пикселей преобразуется в матрицу (двумерный массив) размером 512х512, где каждый её элемент представлен в виде целочисленного значения от до 0 до 255 (яркость белого цвета). Далее эта матрица разбивается на блоки размером 8х8 и укладывается в массив длинной 4096.

Таблица

|  |  |
| --- | --- |
| Исходный размер покрывающего изображения | 512 х 512 |
| Количество каналов покрывающего изображения | 1 (черно/белое) |
| Размер матрицы покрывающего изображения | 512 х 512 |
| Значения элементов матрицы покрывающего изображения | {0, 1, …, 255} |

Встраиваемое изображение также преобразуется в матрицу размером 64х64 , но с бинарными элементами, где 1 — то пиксель закрашен, а 0 — не закрашен.

Таблица

|  |  |
| --- | --- |
| Исходный размер погружаемого изображения | 64 х 64 |
| Количество каналов покрывающего изображения | 1 (черно/белое) |
| Размер матрицы погружаемого изображения | 64 х 64 |
| Значения элементов матрицы погружаемого изображения | {0, 1} |

### 4.1.2 Этап 1. Определение блоков кандидатов

Для каждого полученного блока размером 8х8 вычисляется значение энтропии Шеннона и энтропии предложенной в статье Пал. Сумма обоих значений складывается в итоговое значение. Из всех 4096 блоков выбираются 2048 с наименьшим показателем суммы энтропий. Эти 2048 блоков считаются блоками-кандидатами.

В данной работе, так же как и в исследуемой научной статье, в покрывающее изображение будет погружаться двоичное изображение размером 64х64, то есть 4096 бит. В каждый блок размером 8х8 будет погружаться 4 бита, а это значит, что для погружения всего изображения потребуется 1024 блока. Таким образом, из 2048 блоков-кандидатов нужно будет выбрать 1024 блока, таких чтобы значение целевой функции при этом было минимальным.

Таблица

|  |  |  |
| --- | --- | --- |
| Всего блоков, шт | Блоков-кандидатов, шт | Требуется выбрать, шт |
| 4096 | 2048 | 1024 |

### 4.1.3 Этап 2. Инициализация хромосом генетического алгоритма

Под хромосомой подразумевается одномерный двоичный массив длинной 2048, где 1 означает что блок учавствует в погружении ЦВЗ, а 0 - блок не учавствует, причем единиц (и нулей) в массиве должно быть строго 1024. Таким образом, каждая хромосома представляет собой один из возможных вариантов решения задачи выбора 1024-х блоков.

До начала работы алгоритма, программа случайно генерирует сто таких хромосом. Совокупность этих хромосом называют поколением генетического алгоритма.

Таблица

|  |  |  |
| --- | --- | --- |
| Длина хромосомы | Соотношение 0 и 1 в хромосоме | Количество хромосом |
| 2048 | Строго 1:1 | 100 |

Генетический алгоритм предлагаемый в данной работе включает принципы элитизма. Это означает что на каждой итерации его работы программа будет сохранять некоторое количество лучших решений (хромосом) и сохранять их для формирования следующего поколения.

Максимальное количество итераций алгоритма равно 30.

Таблица. Основные параметры генетического алгоритма

|  |  |
| --- | --- |
| Количество элитных хромосом | Количество итераций алгоритма |
| 20 | 30 |

### 4.1.4 Этап 3. Генетический алгоритм

Первым запускается генетический алгоритм.

Рисунок

# TODO: Добавить блок-схему алгоритма

### 4.1.4.1 Шаг 1. Оценка хромосом и формирование первой элиты

Для каждой хромосомы из первого поколения рассчитывается значение целевой функции. Двадцадь лучших значений хромосом сохраняется в отдельный массив (элита), который будет добавлен в популяцию на следующем круге итерации.

Тут следует отметить что для оценки каждой хромосомы требуется определить параметр *Threshold*. Поскольку его оптимальное значение еще не найдено, то на этом шаге оно берется как среднее внутри возможного интервала, то есть равное 10.

До запуска Шага 2 в генетический алгоритм встраивается алгоритм Светлячков, для определения оптимального значения параметра *Threshold*. Шаг 2 начинает выполняться только после его завершения.

### 4.1.4.2 Шаг 2. Скрещивание и формирование нового поколения

Первое поколение хромосом случайным образом разбиваются на пары и образуют две новые хромосомы такой же длины (2048) с помощью перестановки своих сегментов в определенном порядке.

Рисунок

# TODO: Добавить иллюстрацию образования новых хромосом из родительских

Поскольку распределение нулей и единиц внутри хромосомы хаотично, то нельзя гарантировать что в новых хромосомах их пропорция будет строго 1:1. Экспериметнально выявлено что их соотношение близко к 1:1, поэтому в тех случаях где доля единиц превышает долю нулей, специальная функция случайным образом выбирает нужное количество единиц и заменяет их нулями. И наоборот, в тех хромосомах где больше нулей, некоторые из них заменяются единицами. Такие случайные преобразования хромосом реализуют в предлагаемой имплементации генетического алгоритма появление случайных мутаций, что должно привести к расширению пространства решений в сгенерированной при инициализации популяции.

Для каждой полученной после скрещивания хромосомы считается значение целевой функции.

Таблица. Параметры скрещивания

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Родительские хромосомы, шт | | | Новые хромосомы, шт | | |
| Кол-во | Длина | Пропорция (нулей и единиц) | Кол-во | Длина | Пропорция (нулей и единиц) |
| 100 | 2048 | 1:1 | 100 | 2048 | 1:1 |

Для формирования следующего поколения берется 80 из 100 новых хромосом с наименьшим показателем целевой функции, и 20 сохраненных на предыдущем шаге «элитных» хромосом. Таким образом размерность нового поколения на каждой новой итерации алгоритма не увеличивается, а остается равной 100 хромосом.

Таблица

|  |  |
| --- | --- |
| Состав нового поколения хромосом | |
| 20 элитных хромосом из предыдущего шага | 80 лучших хромосом после скрещивания |

Для всех ста хромосом нового поколения значение целевой функции уже посчитаны, остается лишь вновь выбрать элиту (20 решений с наименьшими показателями целевой функции) и сохранить в отдельный массив.

Шаг 2 повторяется 30 раз. После завершения всех итераций лучшим решением окажется то, которое имеет наименьшее значение целевой функции в массиве элитных хромосом.

### 4.1.5 Этап 3. Алгоритм светлячков

Задача алгоритма Светлячков в предлагаемом техническом решении - найти оптимальное значение параметра *Threshold*. Опытным путем было установлено что оптимальное значение этого параметра находится в пределах от 0 до 20, поскольку прибавление значения 0 никак не изменяет значения частотной области, а значения выше 20 значительно искажают их, что приводит к ухудшению параметров скрытности **SSIM** и **PSNR**. Таким образом задачу оптимизации можносто сформулировать как поиск действительного числа на отрезке от 0 до 20.

Рисунок

# TODO: Добавить блок-схему алгоритма

### 4.1.5.1 Шаг 1. Инициализация популяции светлячков

До начала работы алгоритма, происходит инициализация популяции из 20 особей, каждая из которых будет иметь случайное значение в заданных пределах и представлять собой один из возможных вариантов параметра Threshold.

Таблица

|  |  |
| --- | --- |
| Размер популяции | Пределы значений |
| 20 | {0...20} |

### 4.1.5.2 Шаг 2. Оценка популяции и выбор лучшего кандидата

Для каждой пары светлячков *xi* и *xj* рассчитывается значение целевой функции *f(xi)* и *f(xj)*. Целевая функция здесь, также как и в генетическом алгоритме — это комплексный показатель качества ЦВЗ. Для расчета этих значений с первого шага Генетического алгоритма берется кандидат с лучшим показателем целевой функции.

Для особи с меньшим значением целевой функции (то есть с лучшим показателем качества) рассчитывается значение расстояния *rij*и параметр привлекательности *β(xi)*.

Для особи с большим значением целевой функции, рассчитывается его новое значение (положение на отрезке от 0 до 20) с учетом привлекательности сравниваемого светлячка.

Пример.

Если *f(xi)* < *f(xj),* то для xi вычисляется значения *rij*и *β(xi)*.

Поскольку светлячки в данной задаче живут в одномерном пространстве то расстояние между ними — это просто разность xi и xj:

*r = xi — xj*

Параметр привлекательности зависит от расстояния между светлячками, а также от параметра адсорбции света:

*β(xi) = β0 e-Ɣr2*

где *Ɣ* — параметр адсорбции света равен 0.01, *β0* равен единице.

Новое положение светлячка x будет рассчитано как:

xj new = xjold + *β(xi) + α*

где *α -* это коэффициент инерции для оптимизации сходимости алгоритма. В данной работе этот коэффициент равен 0.05.

Таблица

|  |  |  |
| --- | --- | --- |
| *Ɣ* | *β0* | *α* |
| 0.01 | 1 | 0.05 |

### 4.1.5.3 Шаг 3. Формирование нового поколения

После завершения попарного сравнения всех элементов, значение с меньшей целевой функцией сохраняется отдельно.

Далее так же как на шаге 1 случайно генерируется новая популяция из 20 светлячков, однако случайные три из них заменяются на лучшее значение из предыдущего шага.

Поскольку пространство решений имеет всего одну размерность и ограничего небольшими значениями (от 0 до 20) то алгоритм имеет очень быструю сходимость и оптимальные результаты достигаются уже за 3 поколения. Таким образом шаги 2 и 3 требуется повторить 3 раза, после чего лучший результат передается на второй шаг Генетического алгоритма.