

Векторизация



Векторизация — перевода из растровой графики в векторную, процесс обратный растеризации.

ВОЛНОВОЙ алгоритм

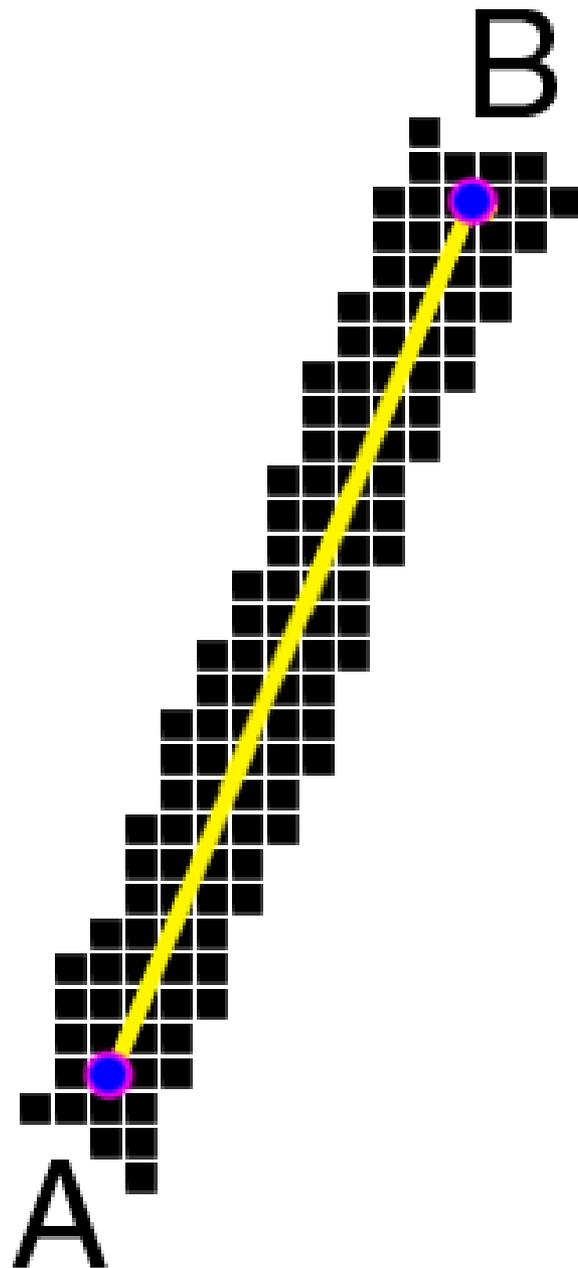


Бинарное растровое изображение

- Под бинарным растровым изображением будем понимать двумерную матрицу из черных и белых точек, в которой объект задается черными точками растра, а фон – белыми точками.

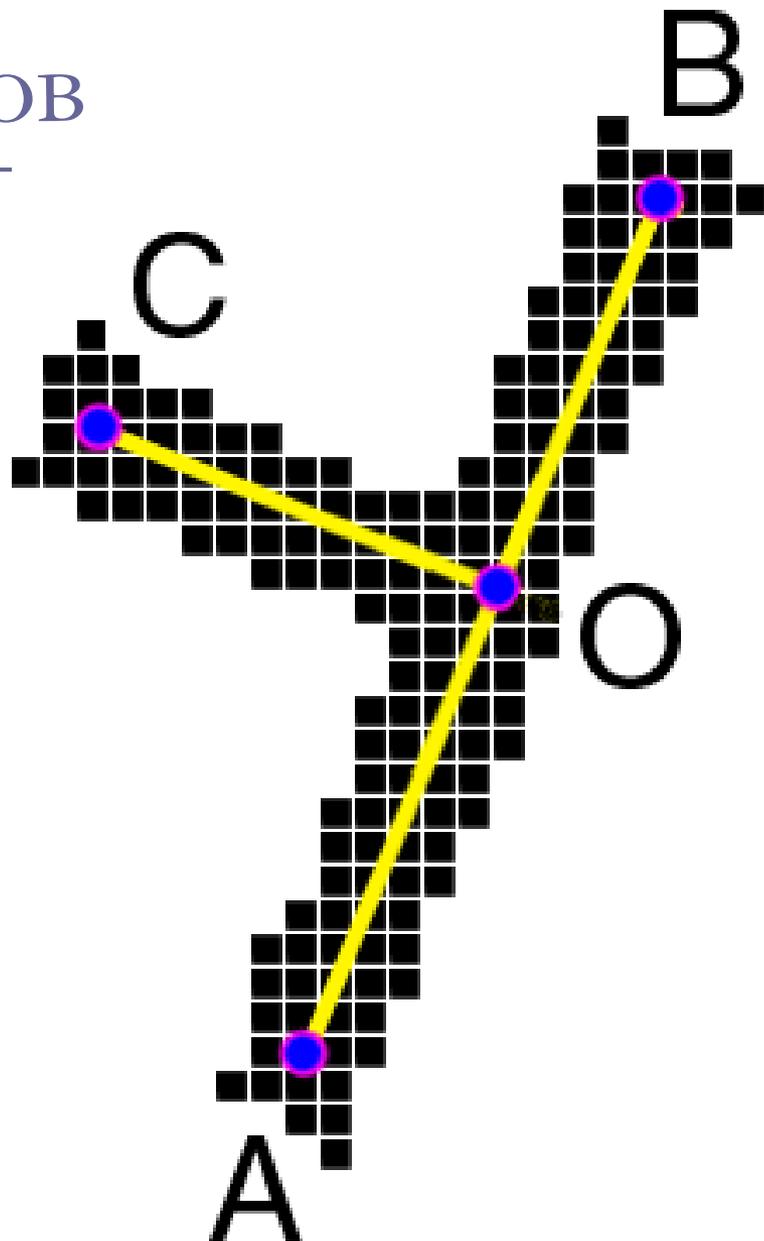
Отрезок

- Под изображением линии (отрезком) на растре будем понимать такое множество черных точек растра, что можно провести отрезок прямой АВ такой, что по обе стороны от этого отрезка будет лежать примерно равное количество точек, и расстояния от отрезка до ближайших крайних точек изображения будут отличаться не более чем на наперед заданную величину



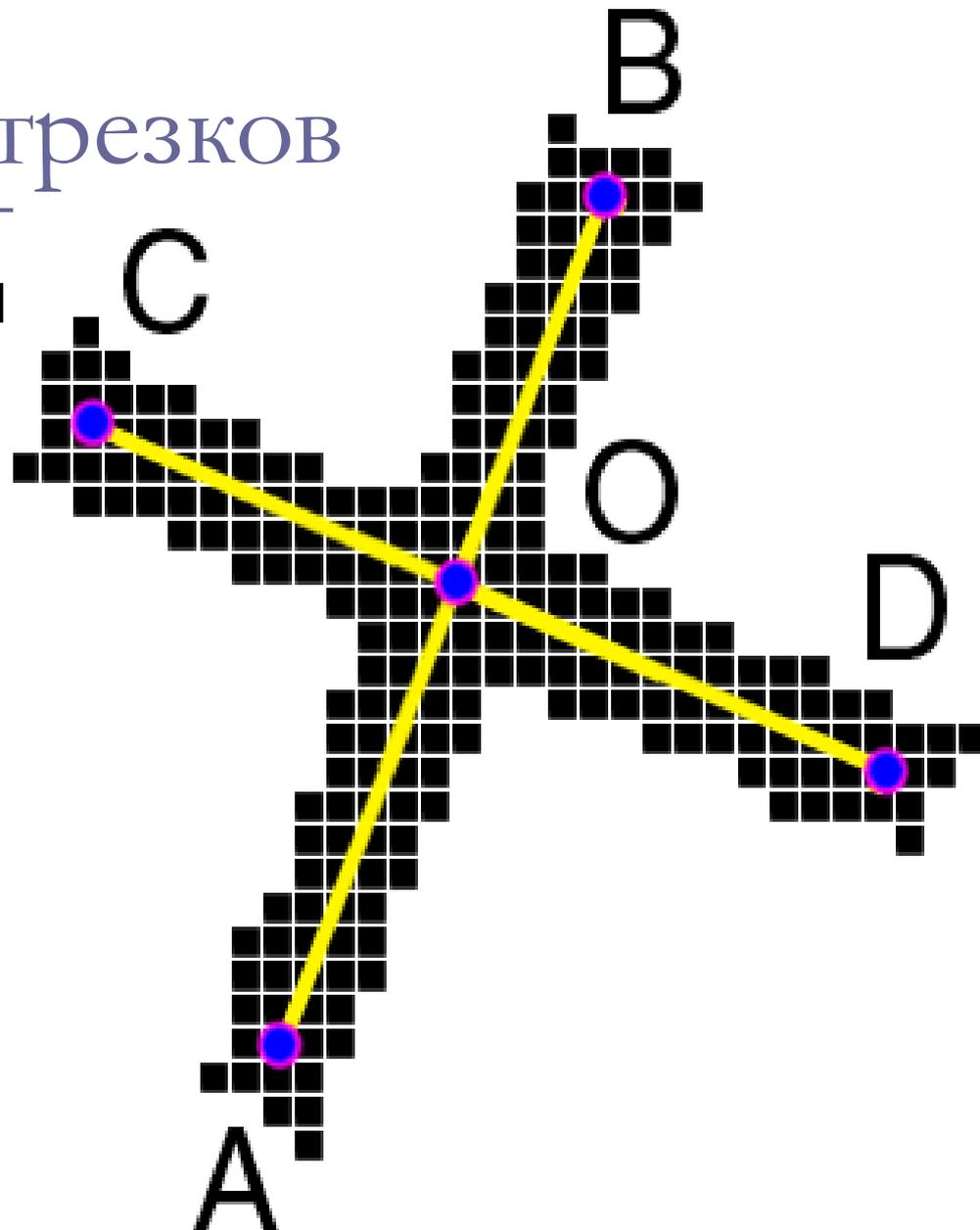
Соединение отрезков

- Под соединением отрезков будем понимать два отрезка такие, что одна из вершин одного отрезка лежит на другом отрезке. В этом случае они будут иметь одну общую точку и один из отрезков будет разделен на два.



Пересечение отрезков

- Пересекающиеся отрезки изображения будем представлять в виде отрезков меньшей длины, имеющих одну общую вершину



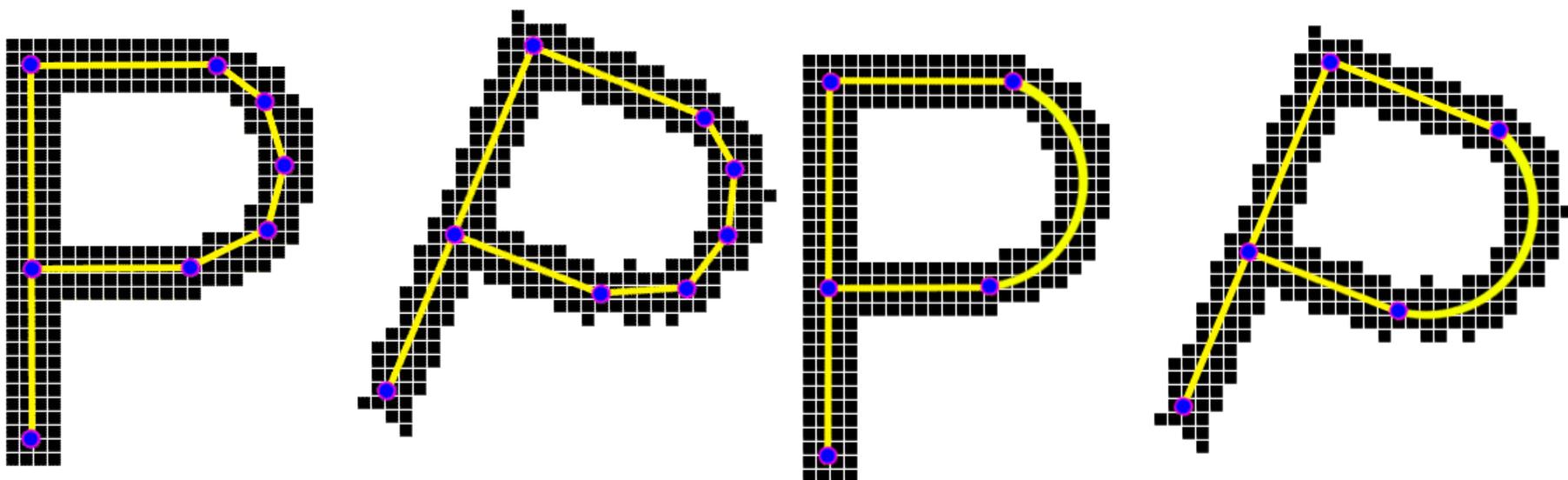
Представление в виде нагруженного графа

- Под векторным представлением растрового изображения будем понимать такой нагруженный граф, у которого нагрузка вершин – пары координат x, y соответствующих узловых точек изображения.
- За узловые точки будем принимать точки соединения изображения линий в растровом изображении. Толщиной линии будем считать ее ширину в пикселях.

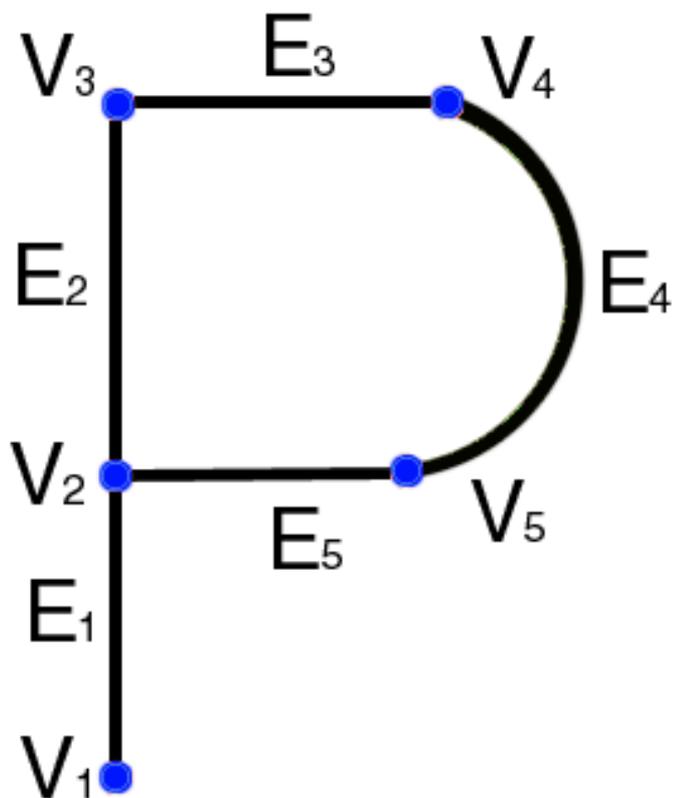
Отрезки и дуги

Изображение может рассматриваться как состоящее из отрезков прямой, либо включающее дуги.

Переход от первого представления ко второму возможен путем нахождения такой последовательности ребер, которая наиболее полно приближается по начертанию к дуге. Для этого необходимо просматривать все последовательности из 3 и более ребер, анализируя взаимное расположение отрезков прямой.



Пример нагруженного графа



Вершина	X	Y	Ребро	Тип
V1	2	2	E1	Прямая
V2	2	14	E2	Прямая
V3	2	29	E3	Прямая
V4	17	29	E4	Дуга R=17
V5	14	14	E5	Прямая

Нагрузка вершин и ребер

Матрица инцидентности

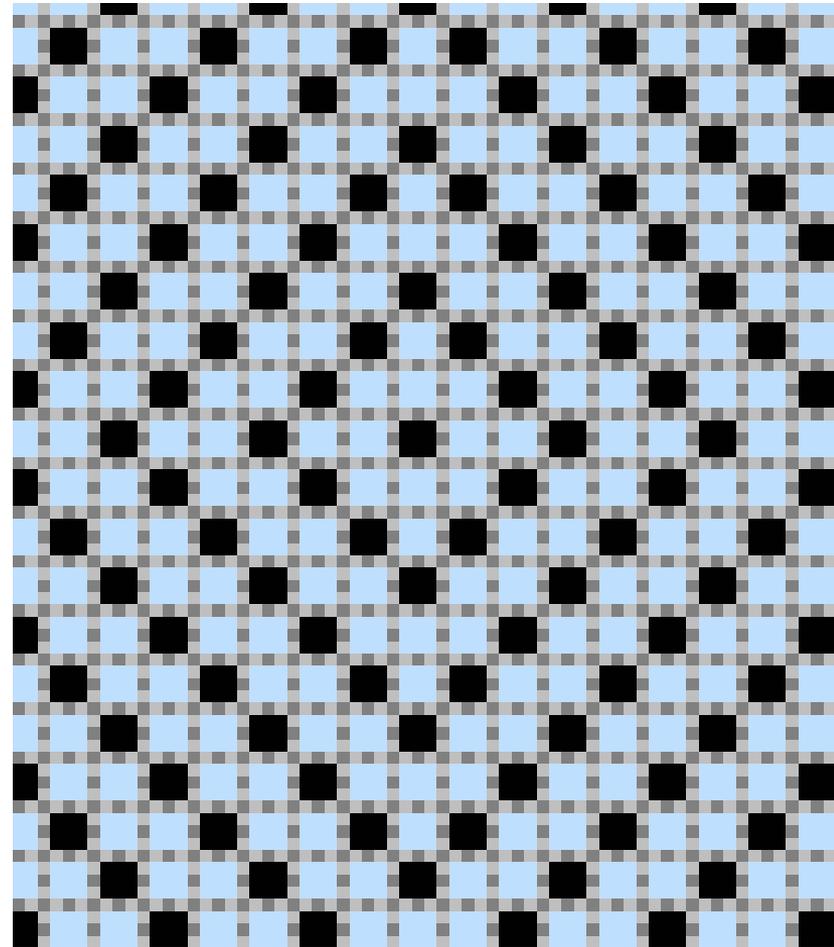
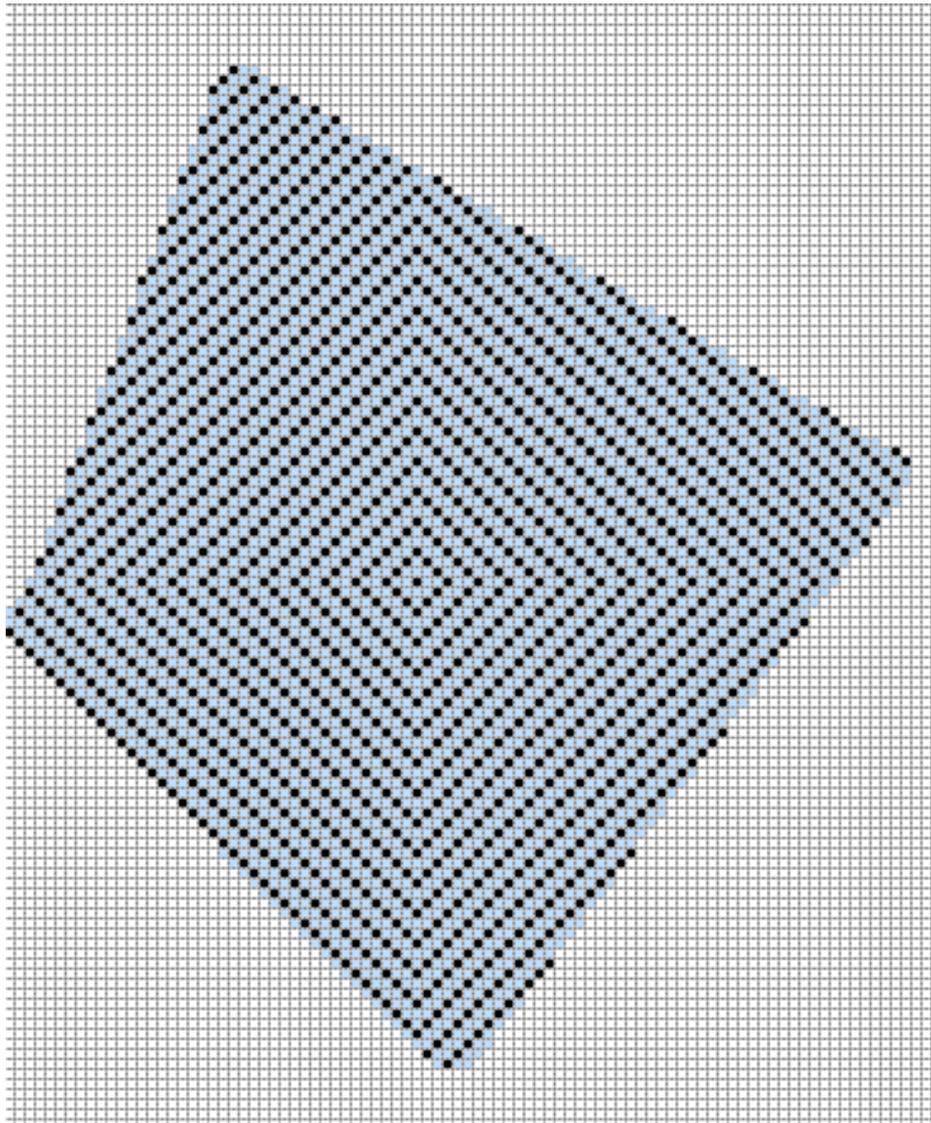
	E1	E2	E3	E4	E5
V1	1				
V2	1	1			1
V3		1	1		
V4			1	1	
V5				1	1

Как видно, по графу возможно восстановить исходное растровое изображение. Для этого необходимо произвести отрисовку ребер графа согласно нагрузкам вершин и матрице инцидентности.

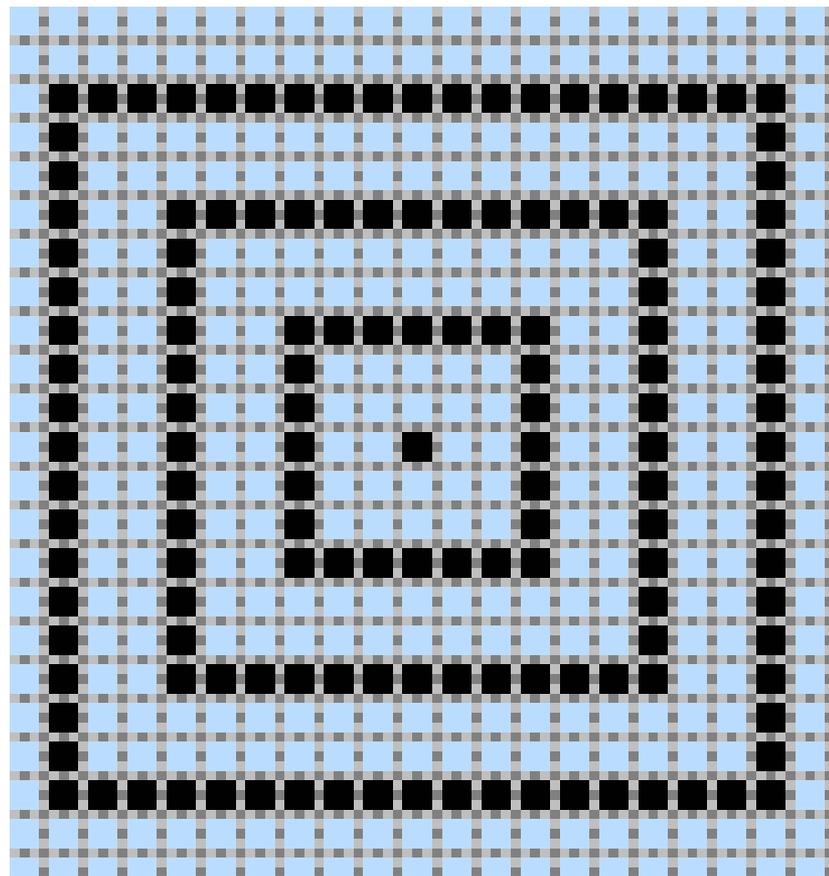
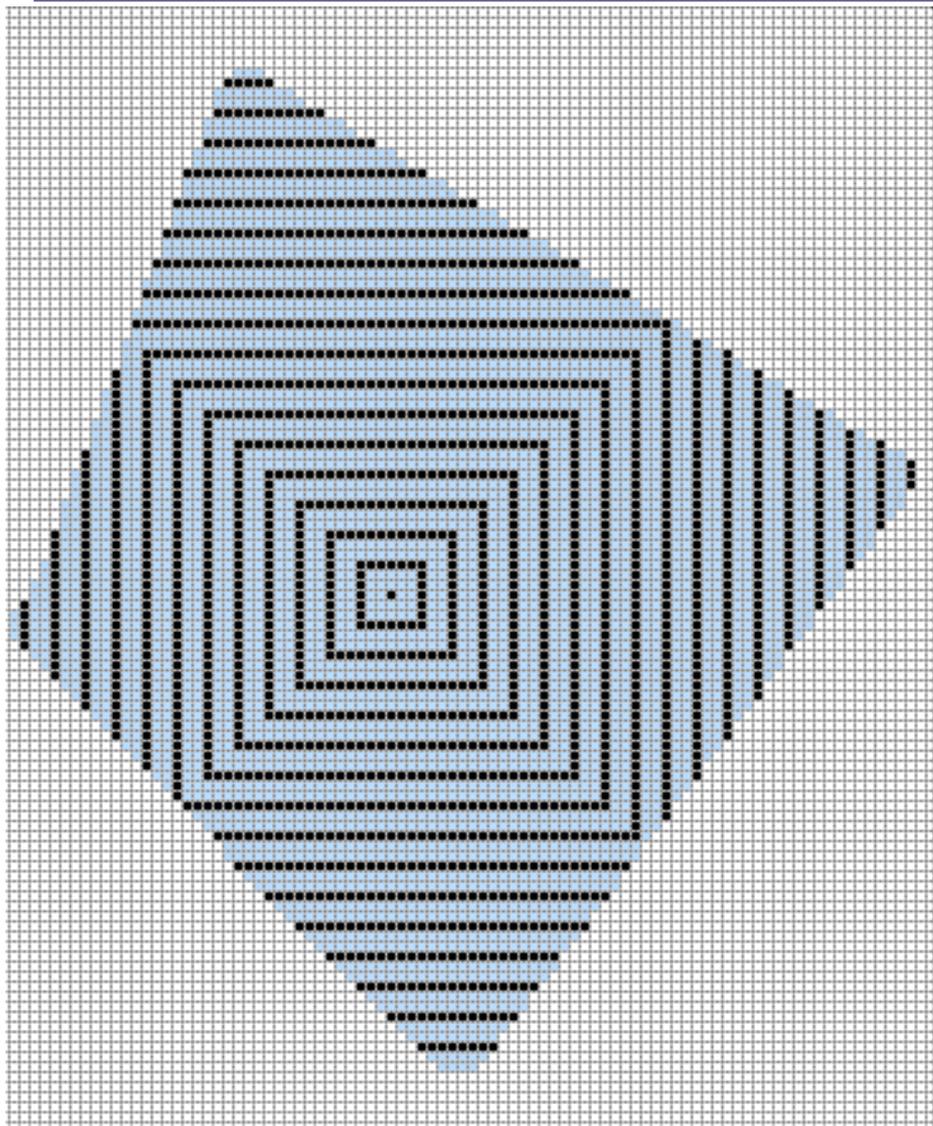
Этапы волнового алгоритма

- Построение скелета изображения с помощью сферической волны
- Оптимизация полученного скелета

Сферическая волна для 4-х связанного растра

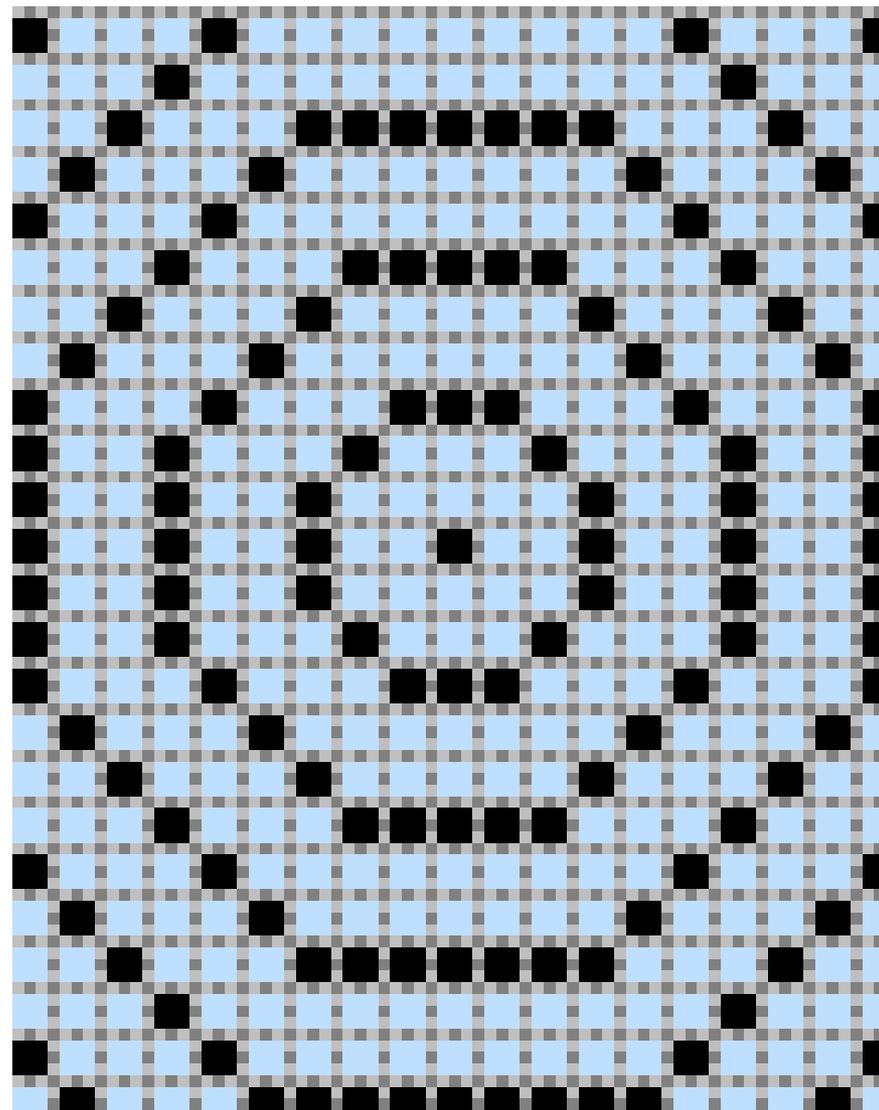
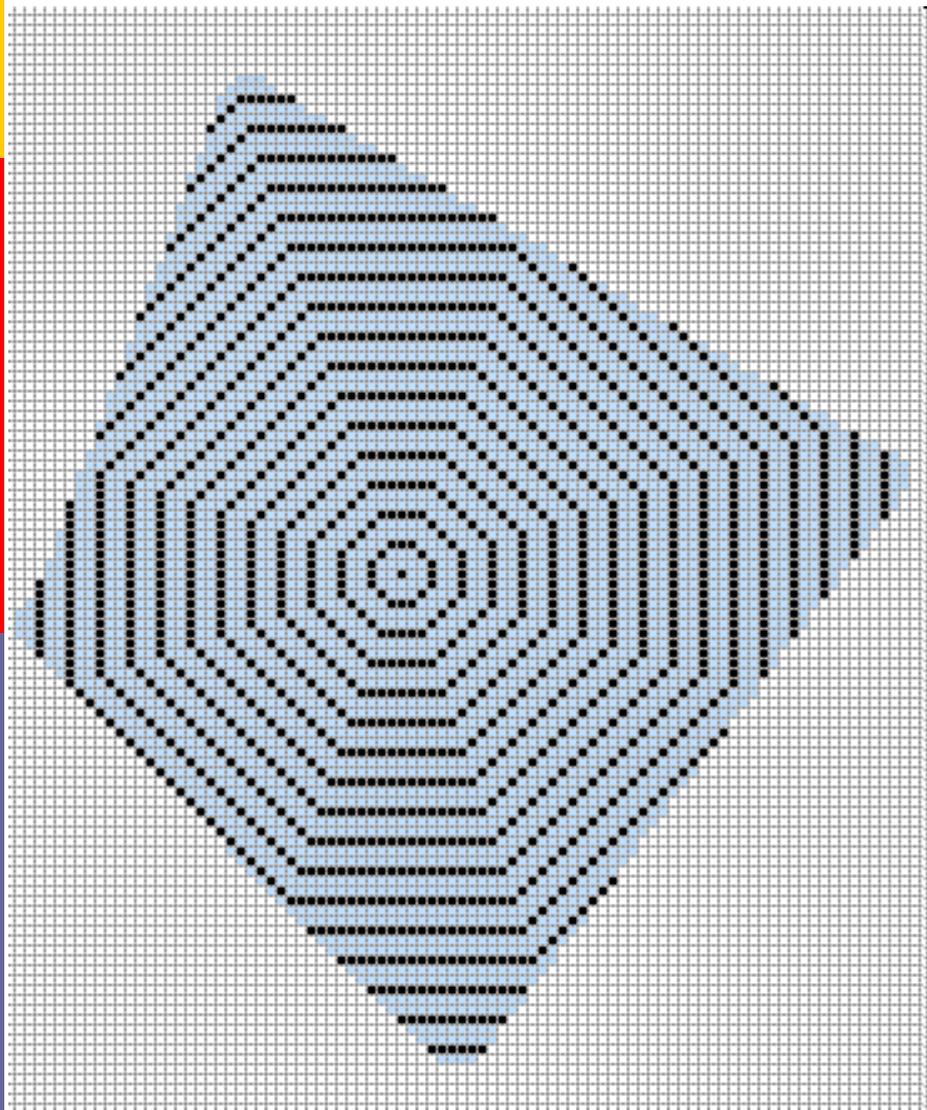


Сферическая волна для 8-ми связанного растра



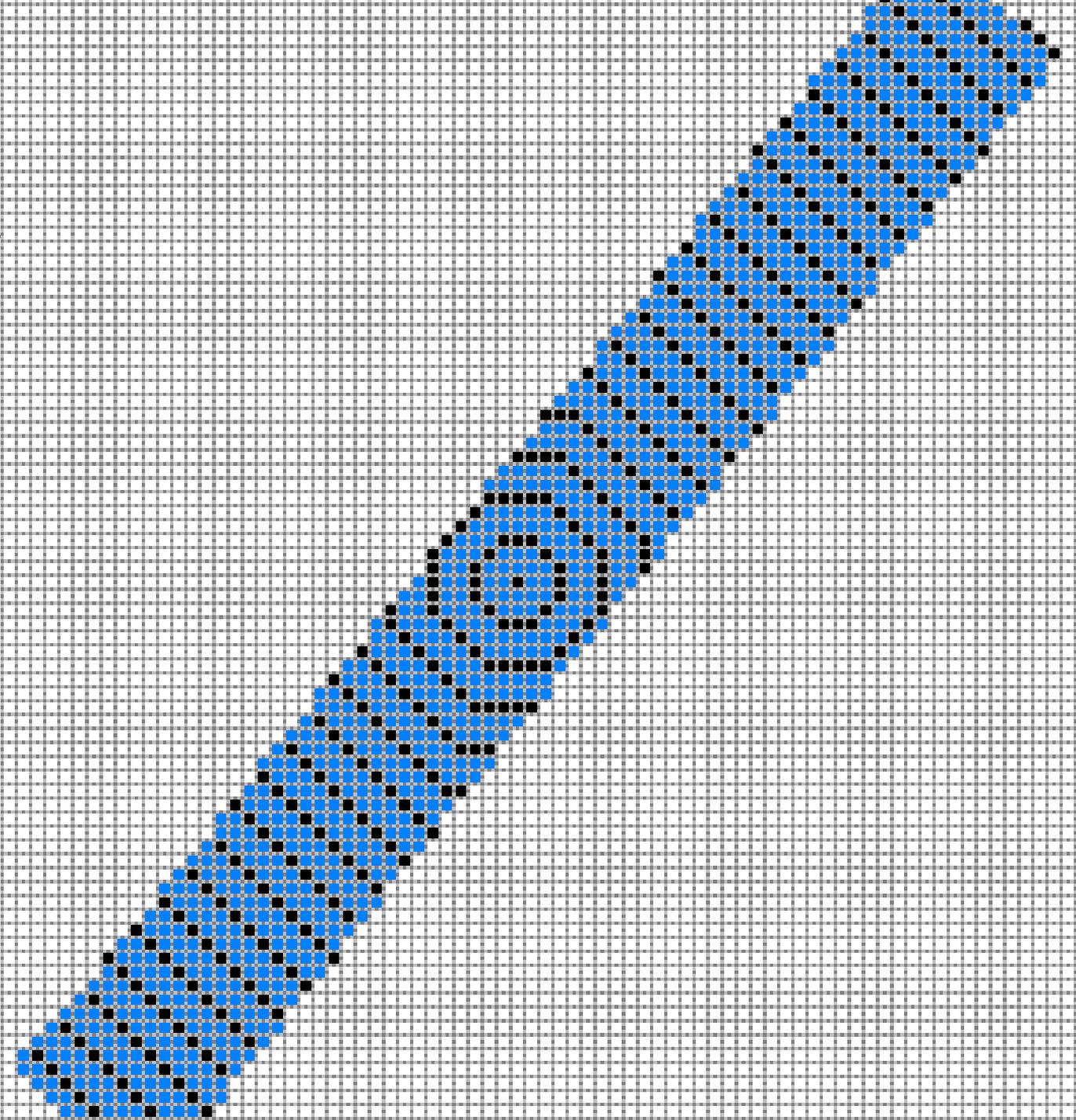
Комбинированная волна

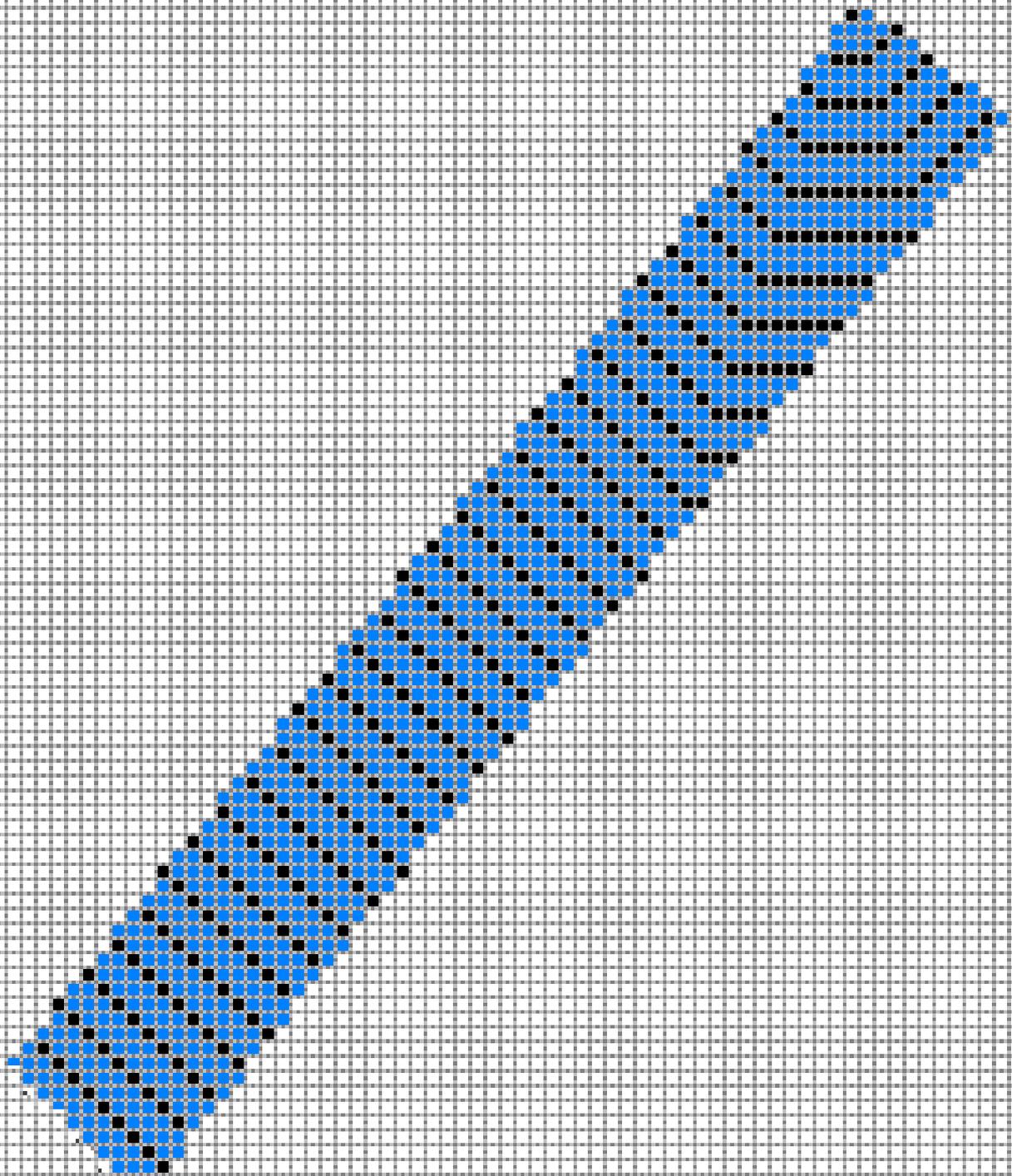
- достигается попеременным применением 4-х и 8-и связного распространения

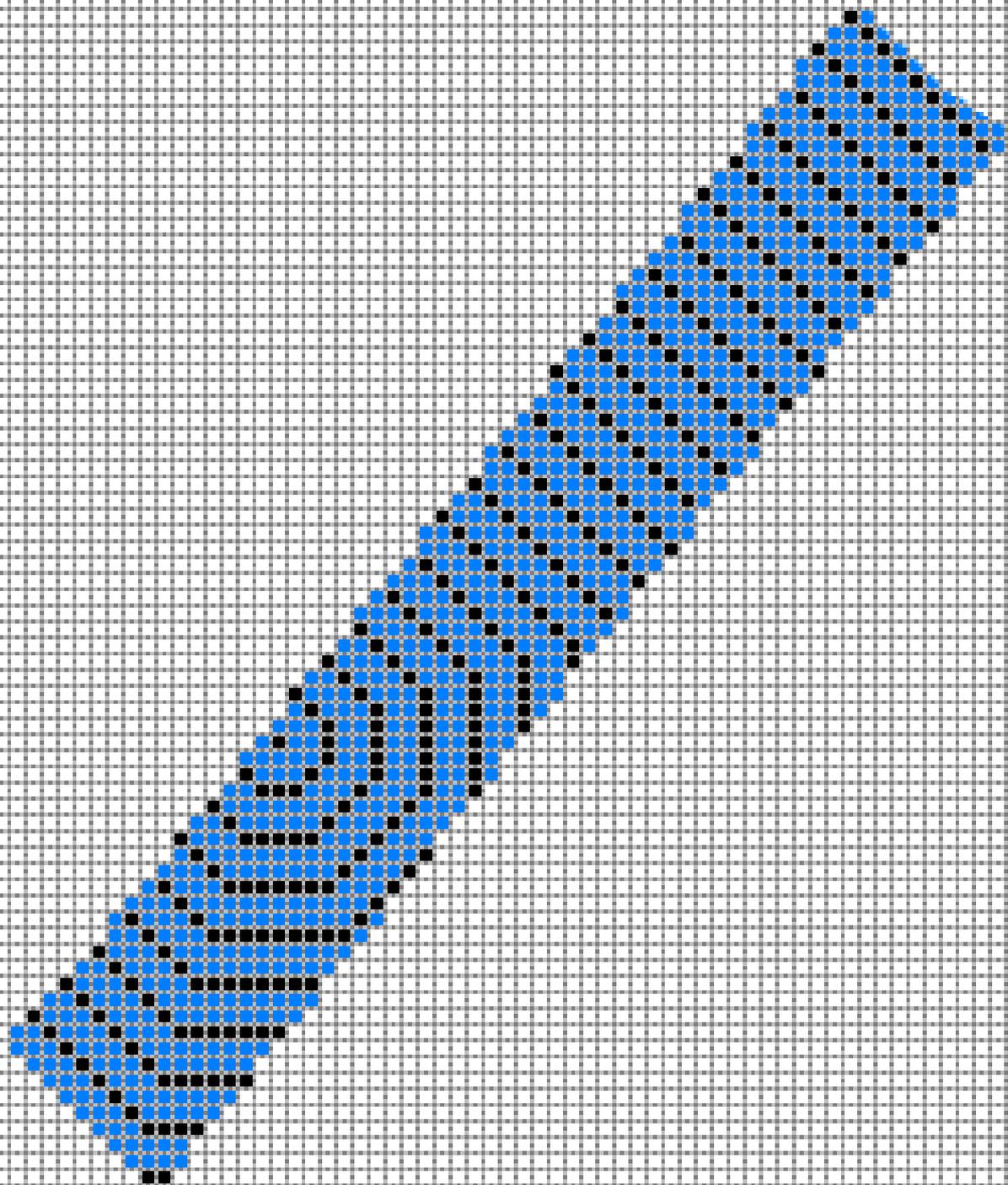


Распространении волны на отрезке

- При распространении сферической волны на отрезке прямой наблюдаются следующие эффекты: не более чем через $2*N$ шагов распространение волны приобретает устойчивый характер вне зависимости от начальной точки распространения волны. Причем N – ширина линии в пикселях.

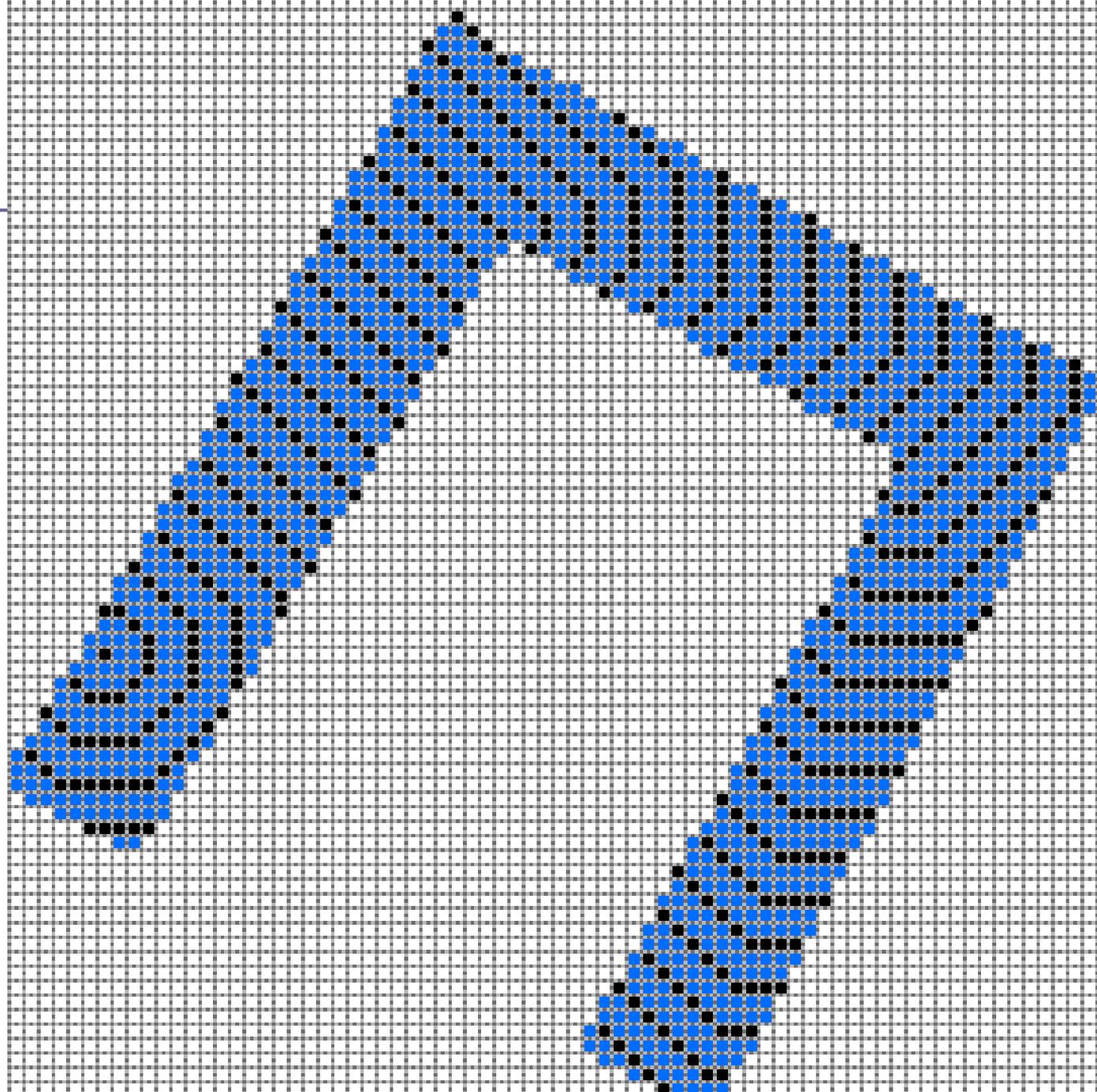


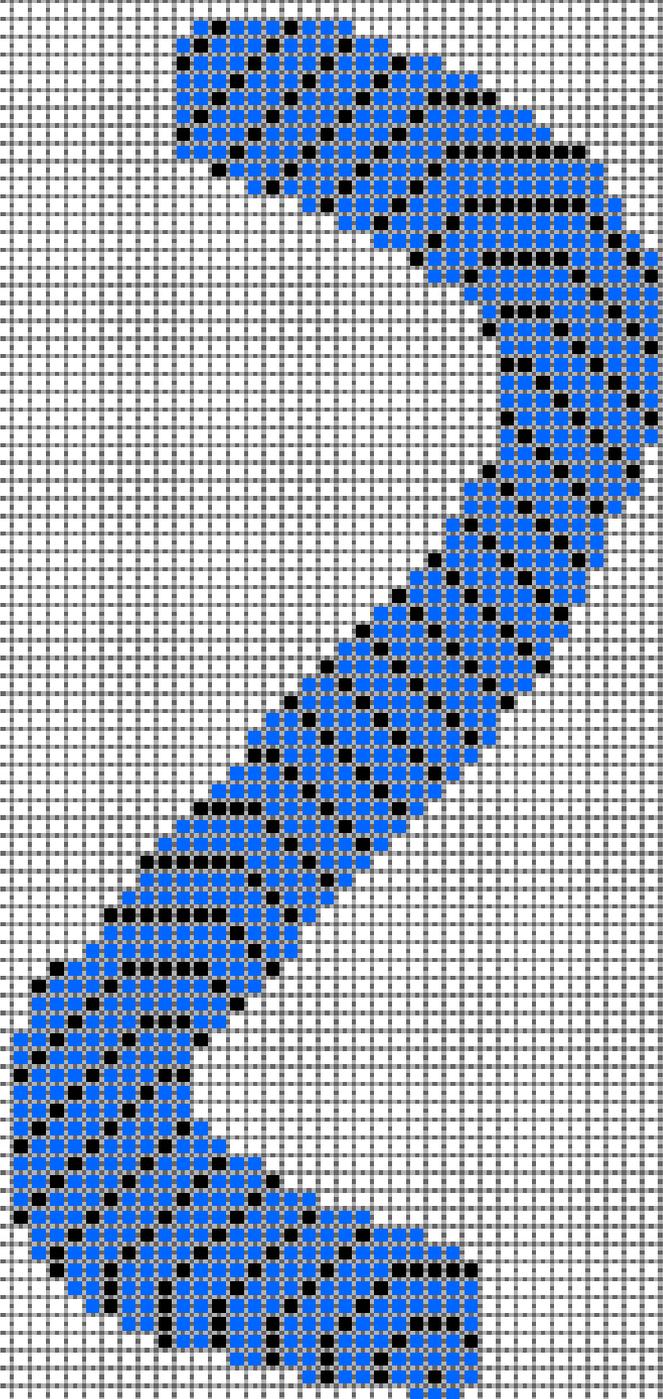




Распространение волны на кривой

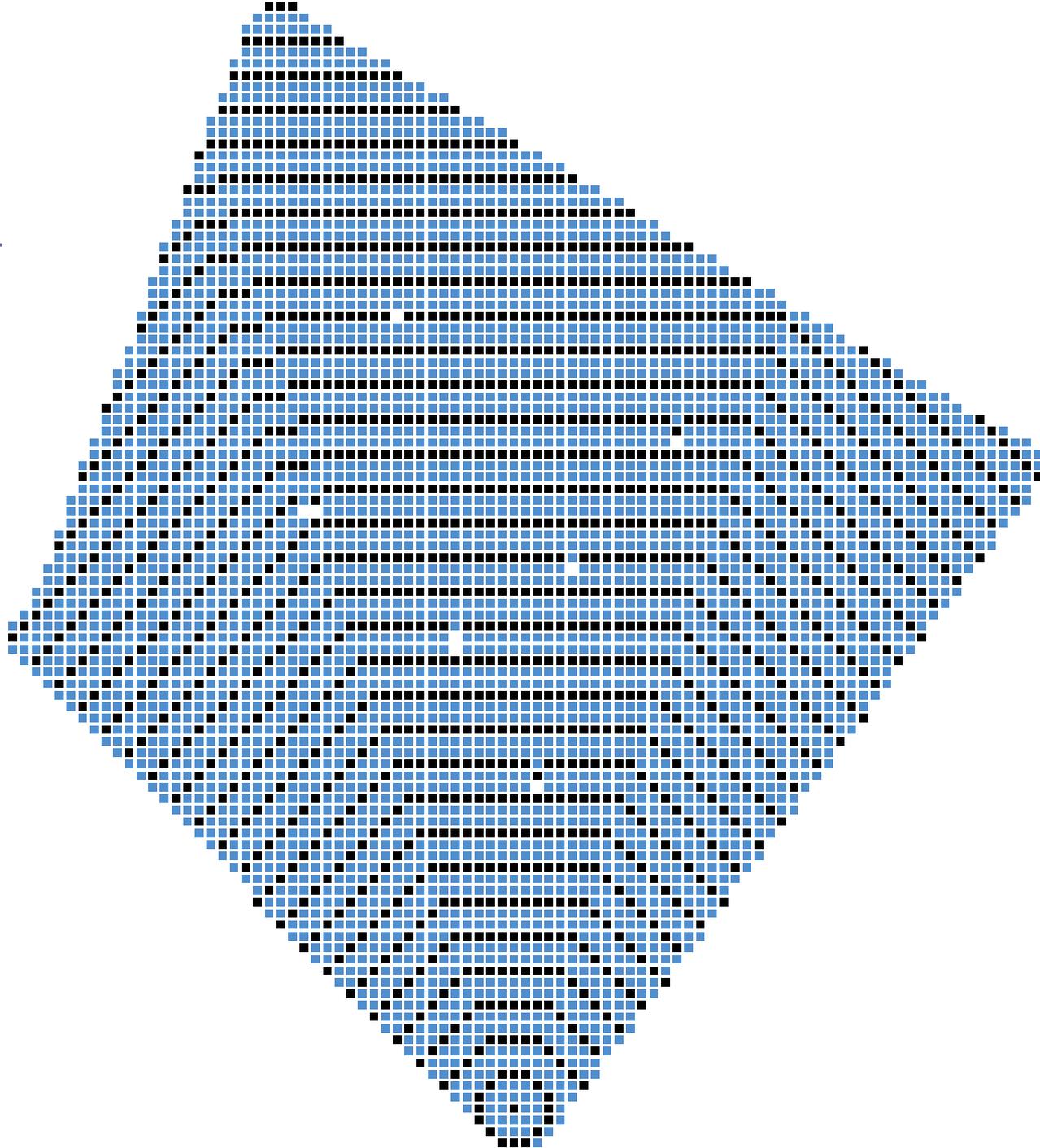
- При отличии формы отрезка от прямой, распространение волны также предсказуемо, причем необходимо отметить хорошие огибающие свойства сферической волны

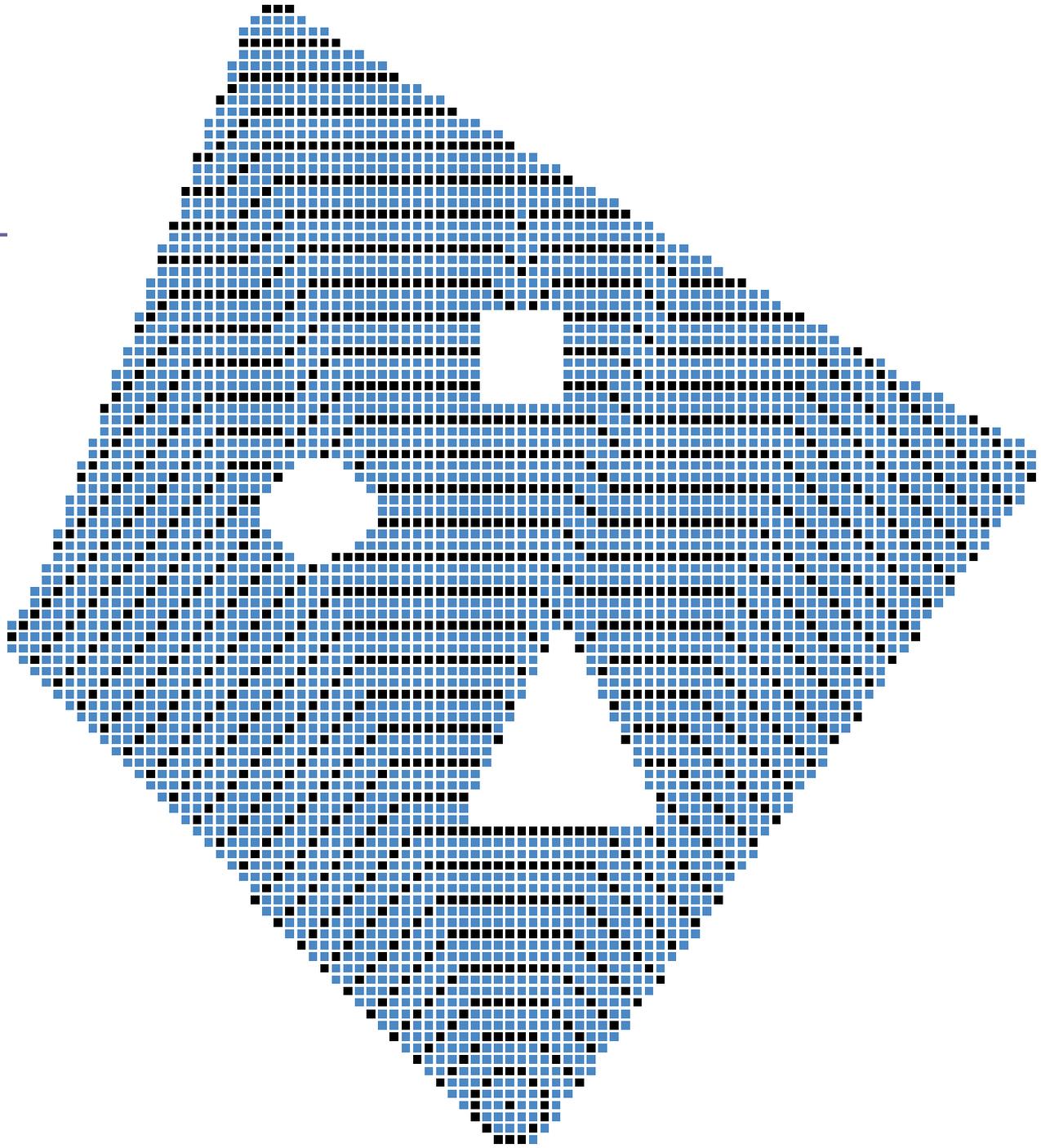




Огибание препятствий волной

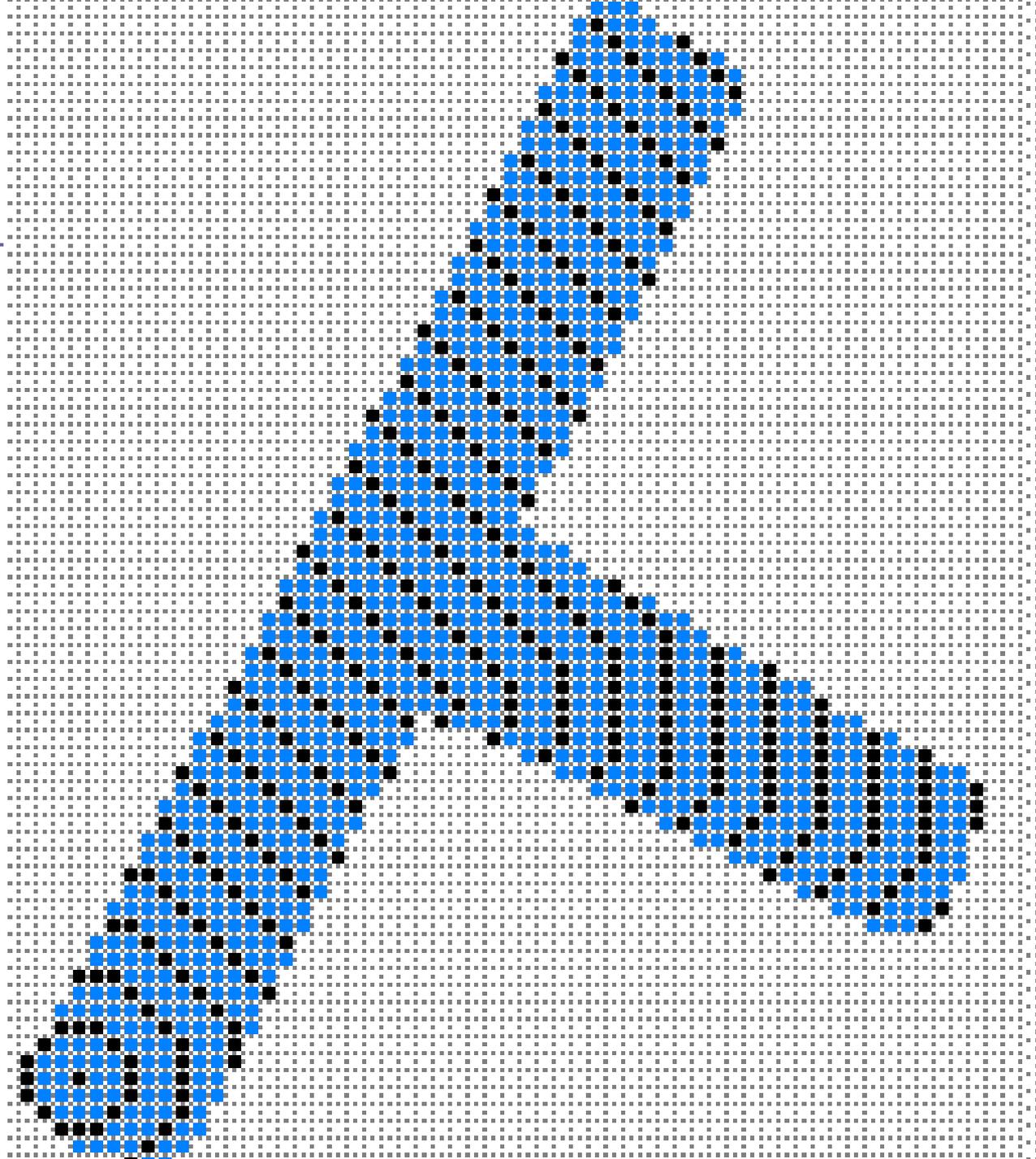
- Если на пути распространения волны встречаются препятствия, то поведение волны целиком зависит от формы и размеров препятствий. Мелкие препятствия (1-2 пикселя) мало влияют на распространение волны, внося незначительные помехи. Более крупные помехи на изображении вызывают значительные искажения картины распространения волны.

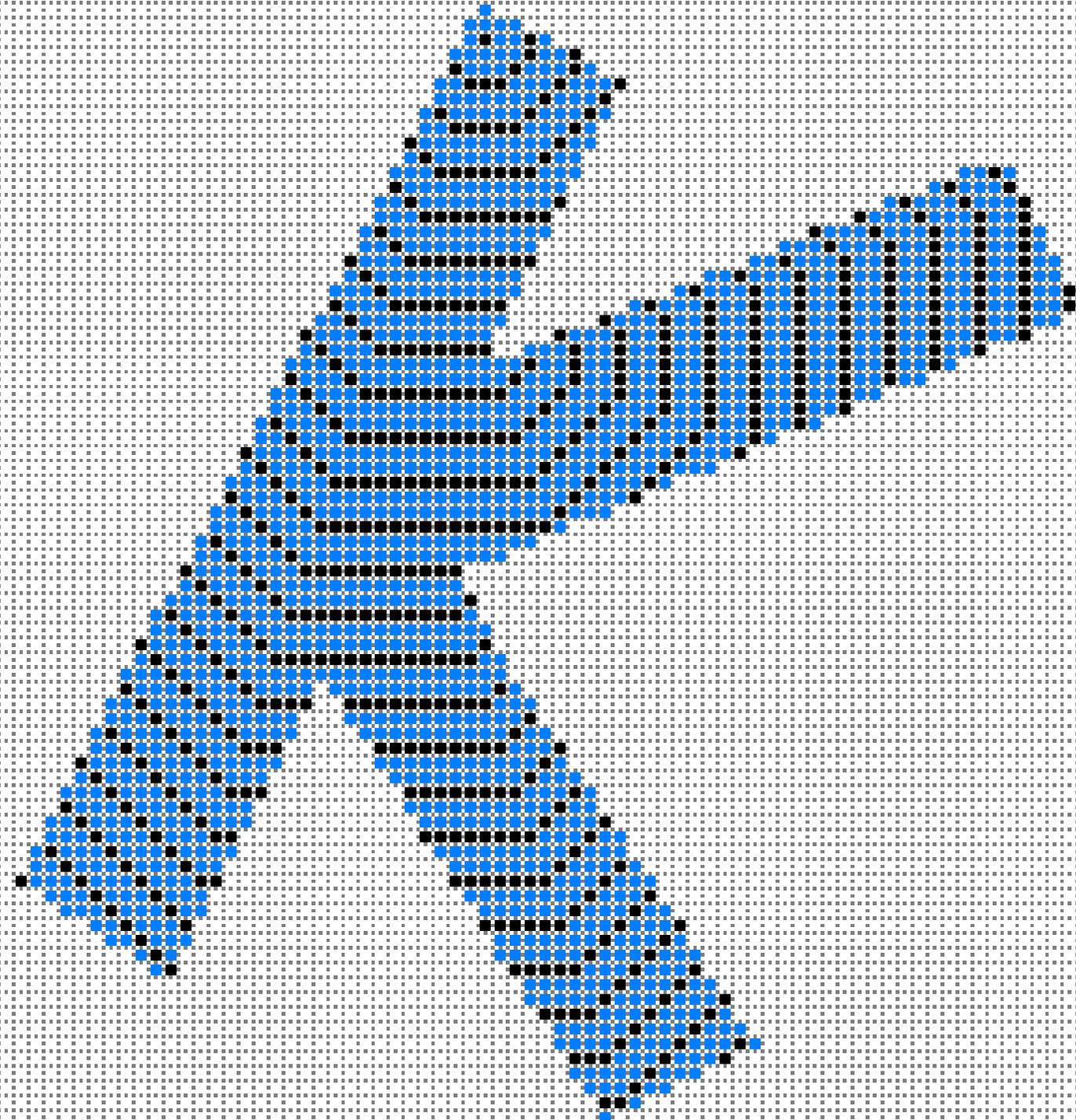


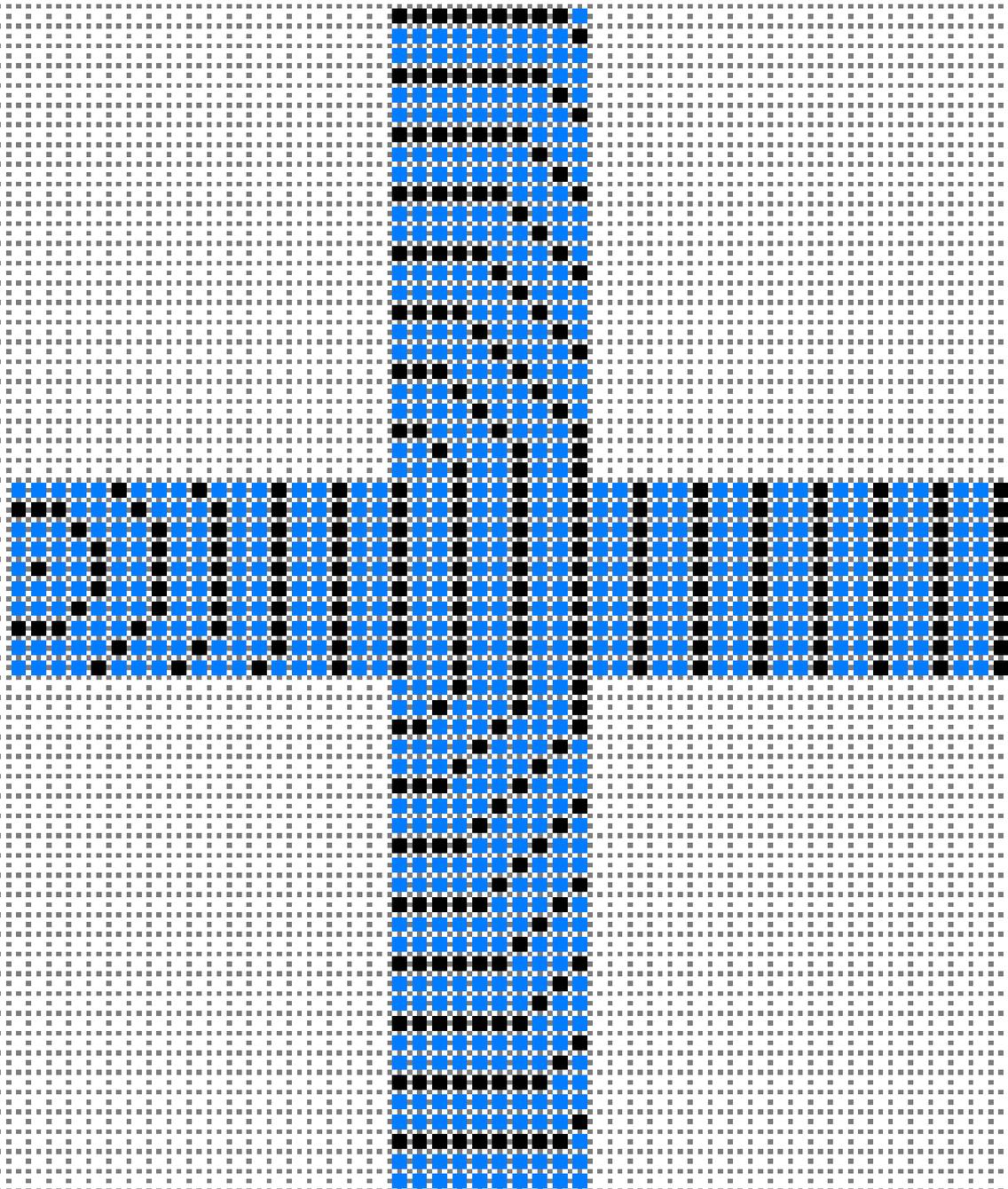


Разделение волны на пересечении отрезков

- При достижении волной места соединения двух или более отрезков наблюдается разделение волны на несколько дочерних волн, сохраняющих поведение материнской волны.
- Момент разделения довольно просто отслеживается путем анализа “ширины” волн, т.е. количества точек образующих очередную генерацию волны: перед разделением наблюдается увеличение “ширины” волны с дальнейшим разделением волны на две (иногда более) дочерние волны.

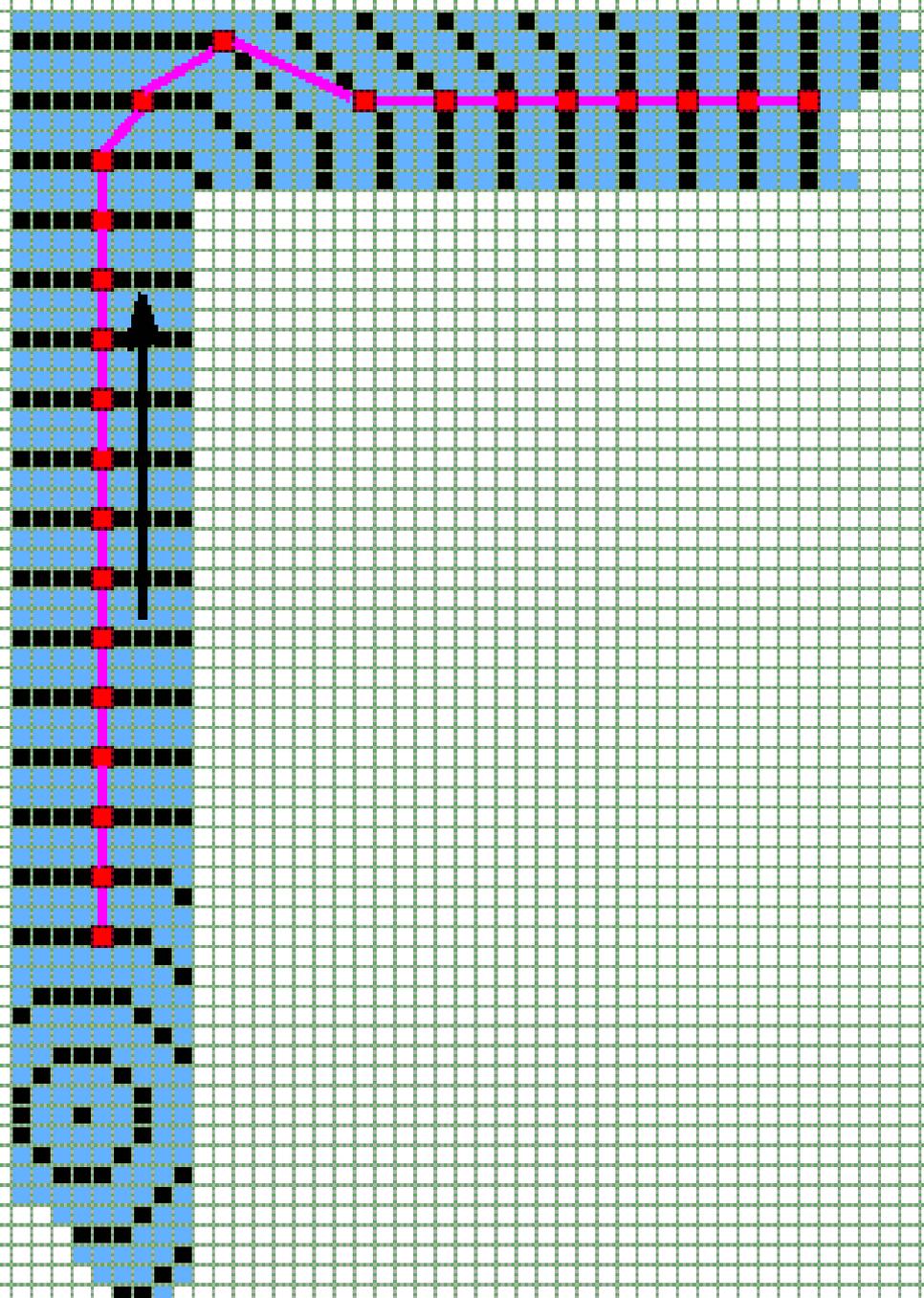


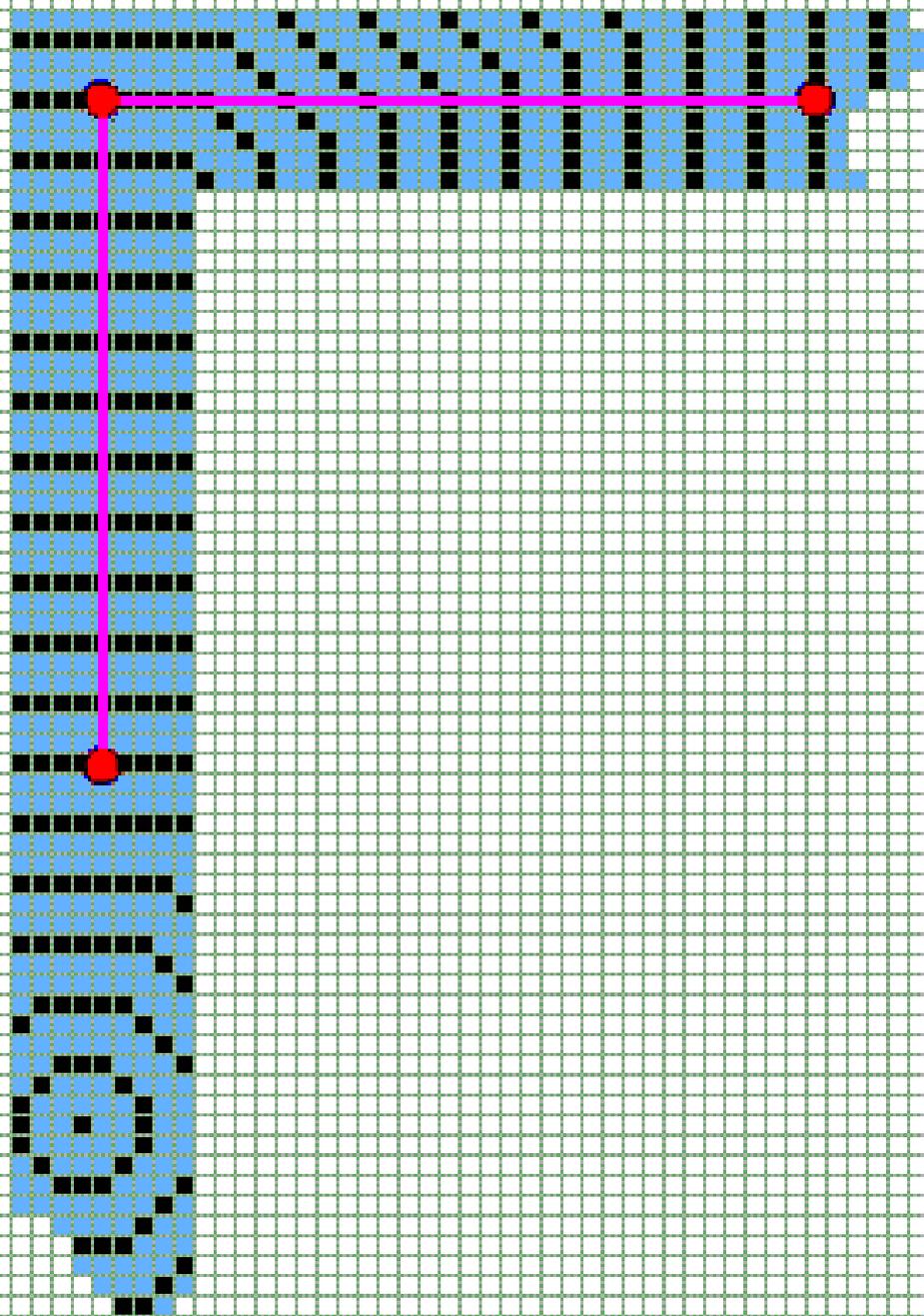




Отслеживание линий изображения

- Отслеживание отрезков производится путем отслеживания перемещения центра отрезка, образуемого крайними точками генерации волны.
- После отслеживания возможно сглаживание отрезков (с целью сокращения узловых точек).
- Если отслеживаемая линия не является прямой, то получается не одна линия, а множество отрезков, интерполирующих исходный рисунок.

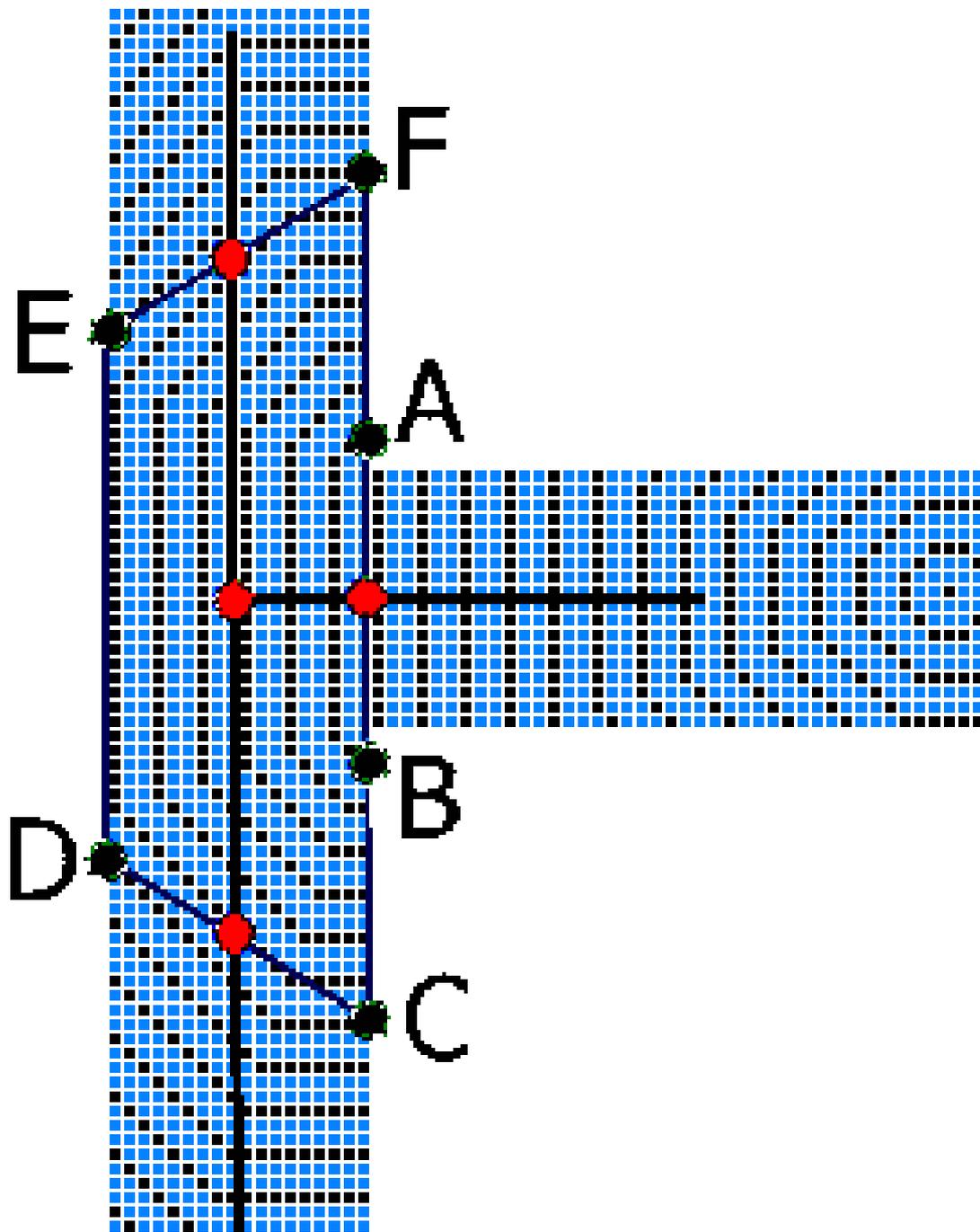




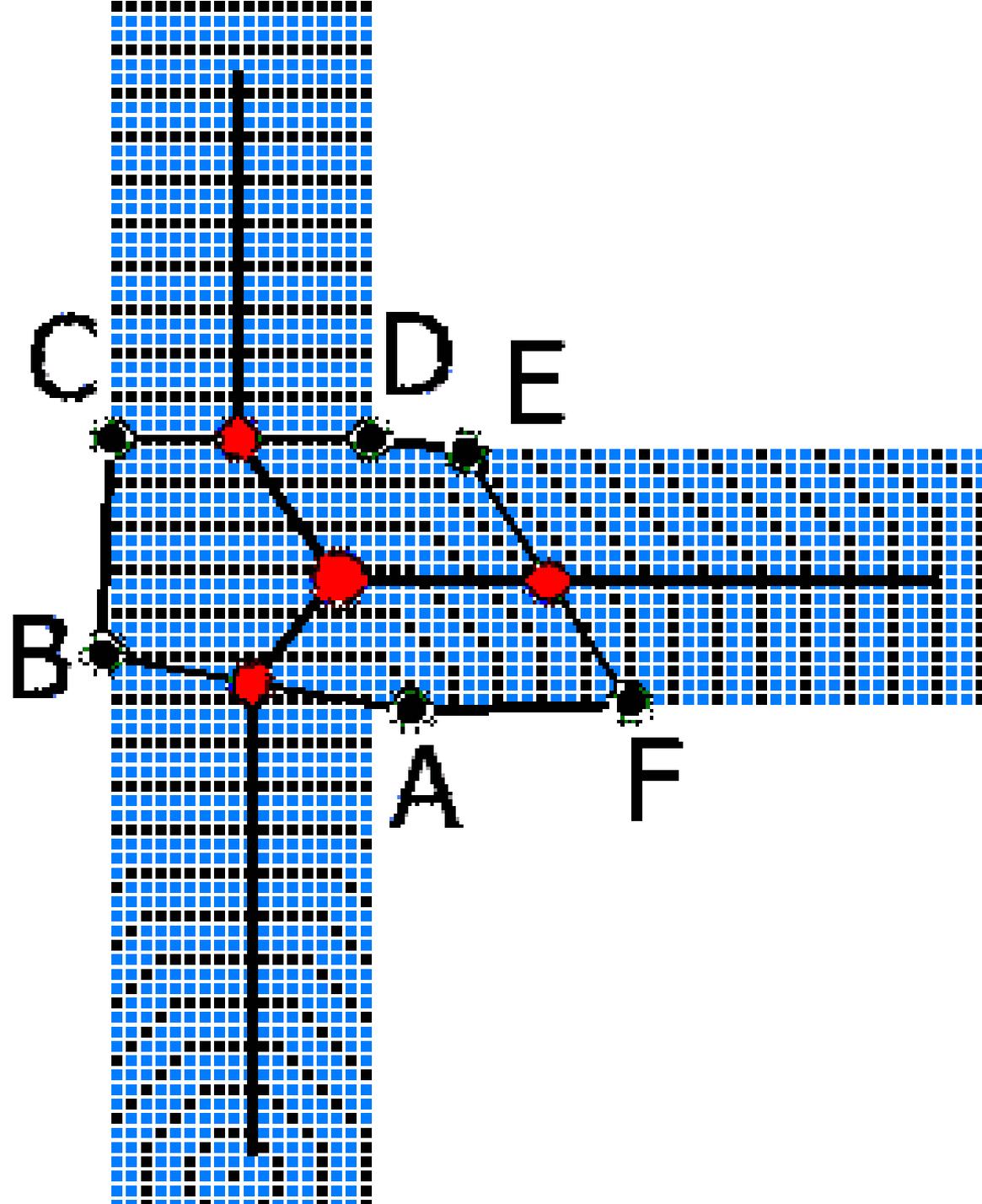
Определение места соединения отрезков

- Выявление мест увеличения “ширины” волны и разделения волны на дочерние позволяет установить точку предполагаемого соединения двух отрезков.
- Определение увеличения “ширины” волны производится путем сравнения “ширины” очередной генерации волны и ее среднего значения за N предыдущих генераций.
- При этом мы получаем 2 крайние точки (A,B) трассируемого отрезка. После разделения волны на 2 полуволны, мы получаем еще 2 пары точек (C,D) и (E,F).
- С помощью анализа этих отрезков (AB, CD, EF) можно оценить положение точки соединения отрезков.

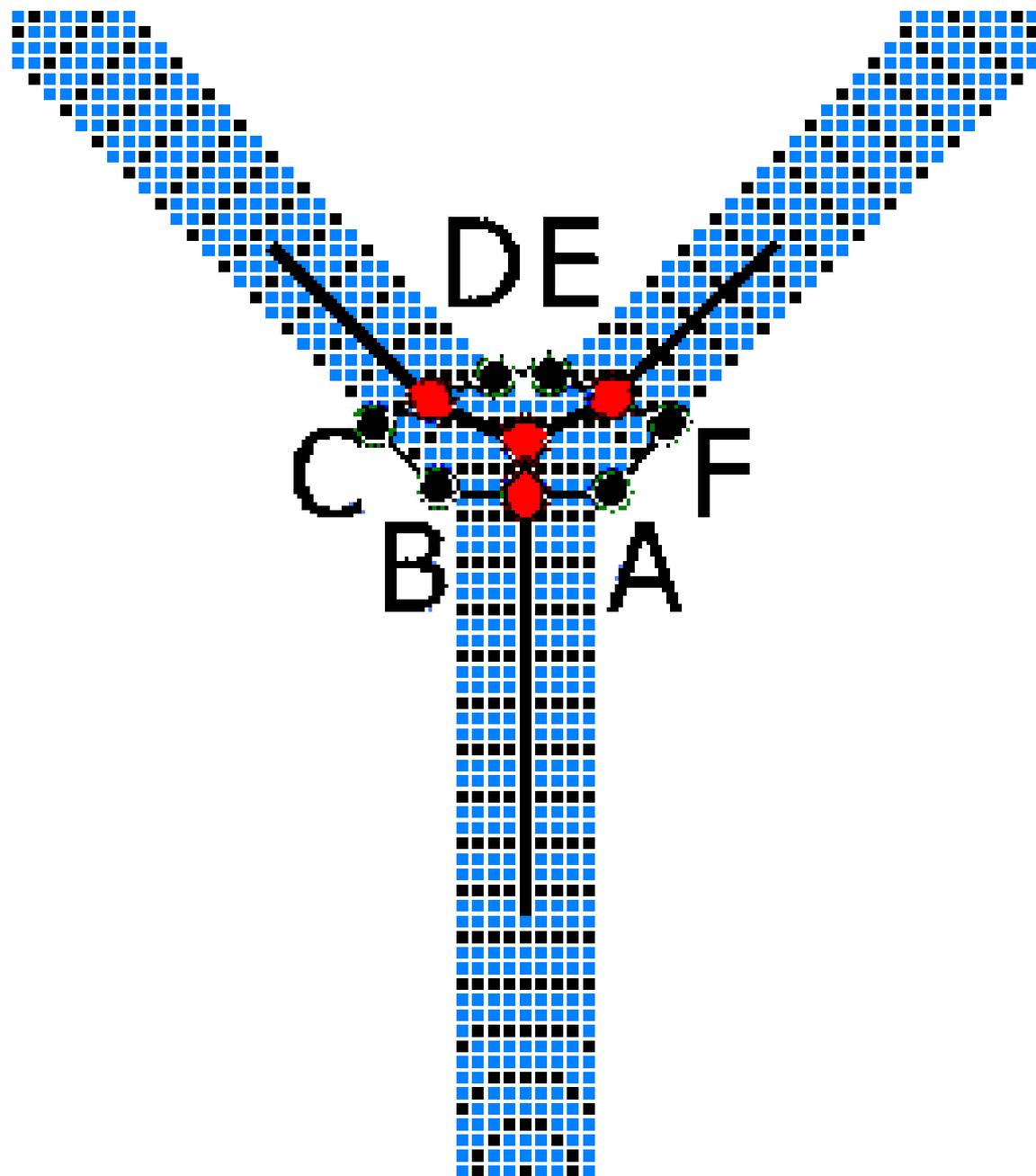
1 вариант



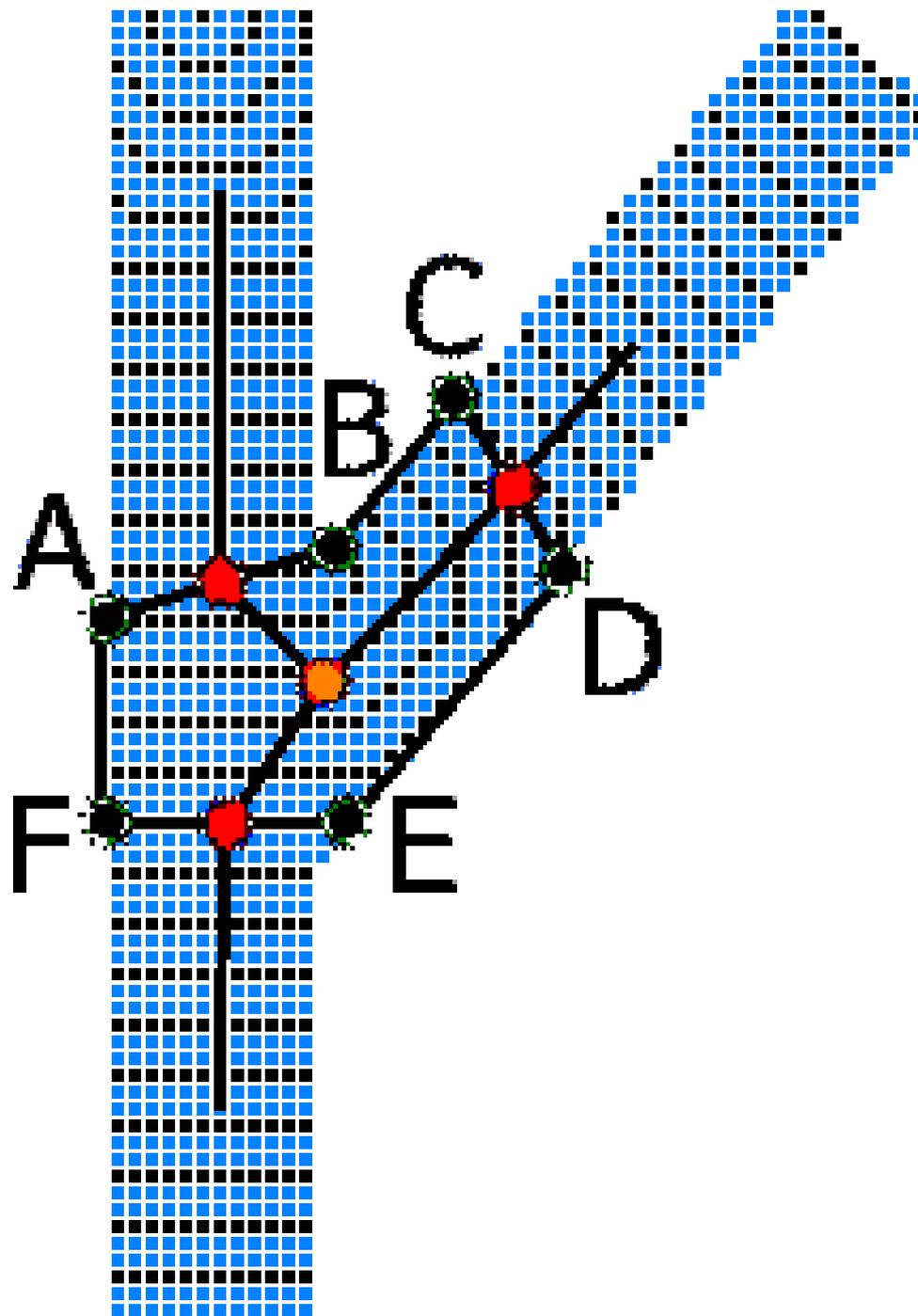
2 вариант



3 вариант

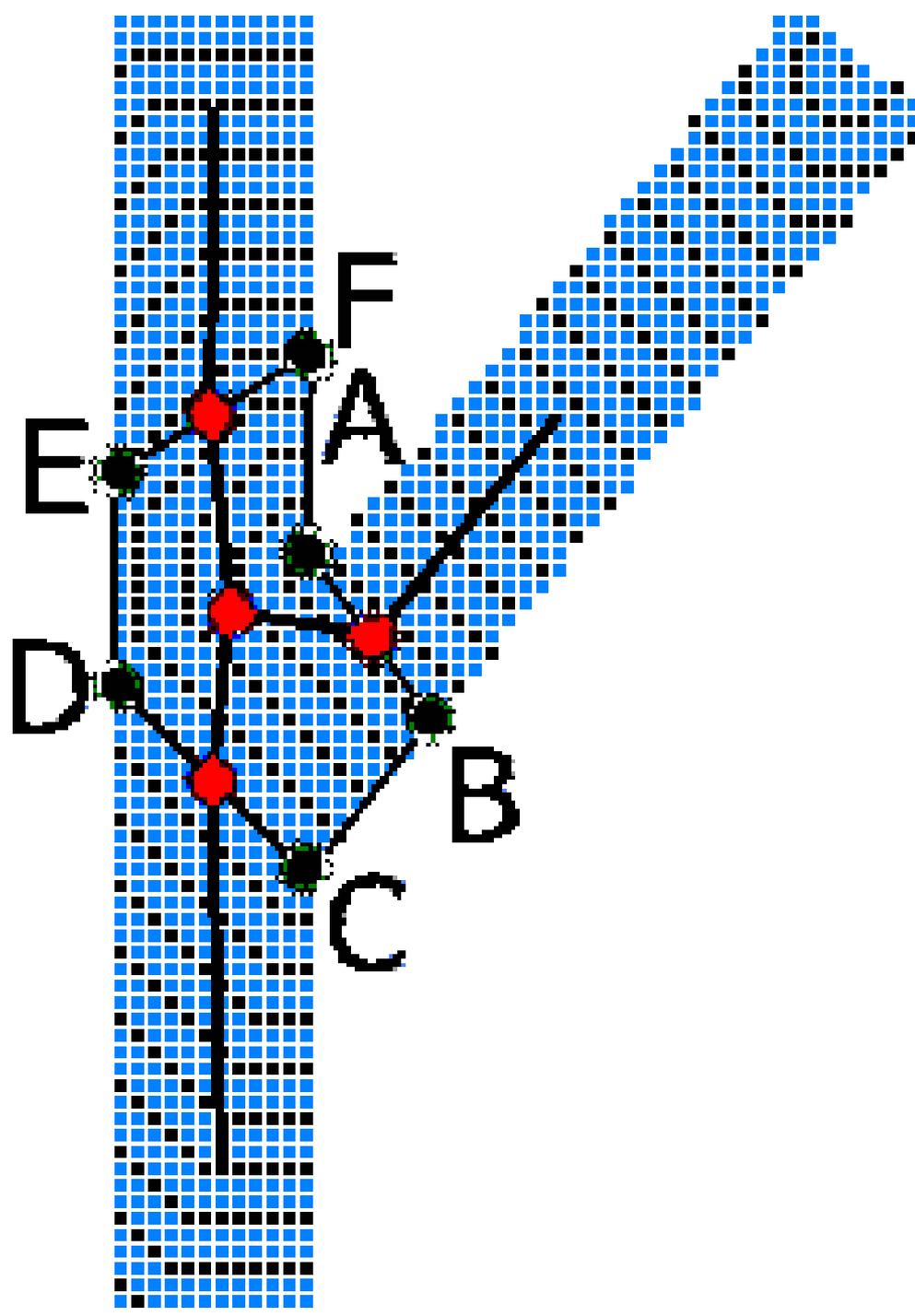


5 вариант



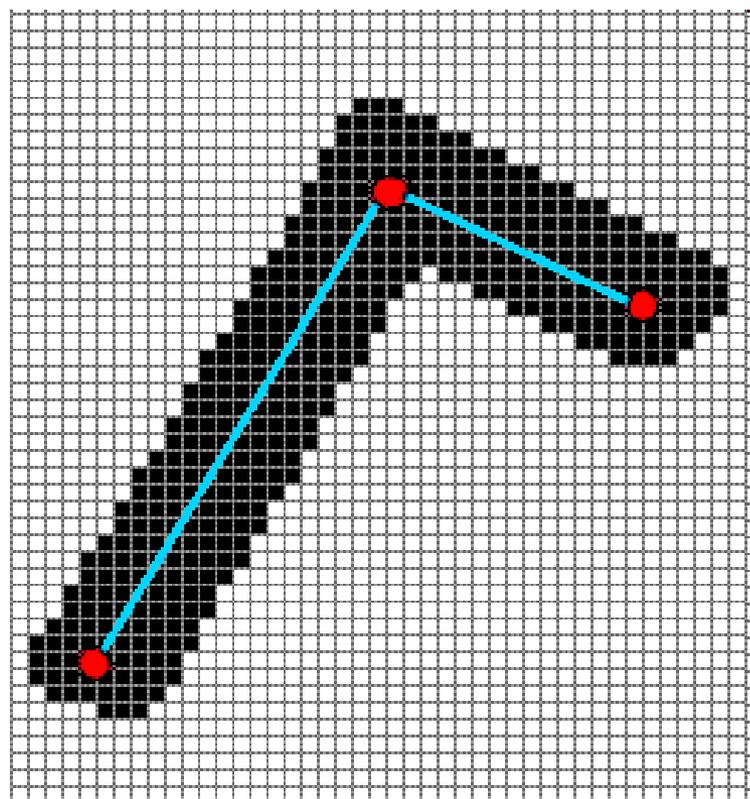
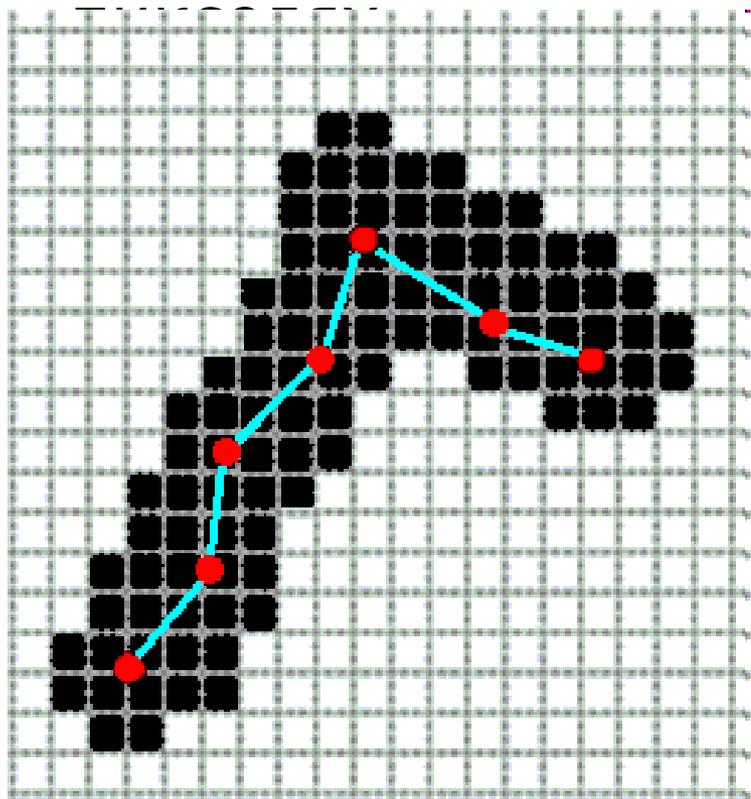
6 вариант

- В любом из возможных вариантов точка соединения отрезков лежит внутри шестиугольника ABCDEF.
- Первоначально установим место соединения как центр масс этого многоугольника.



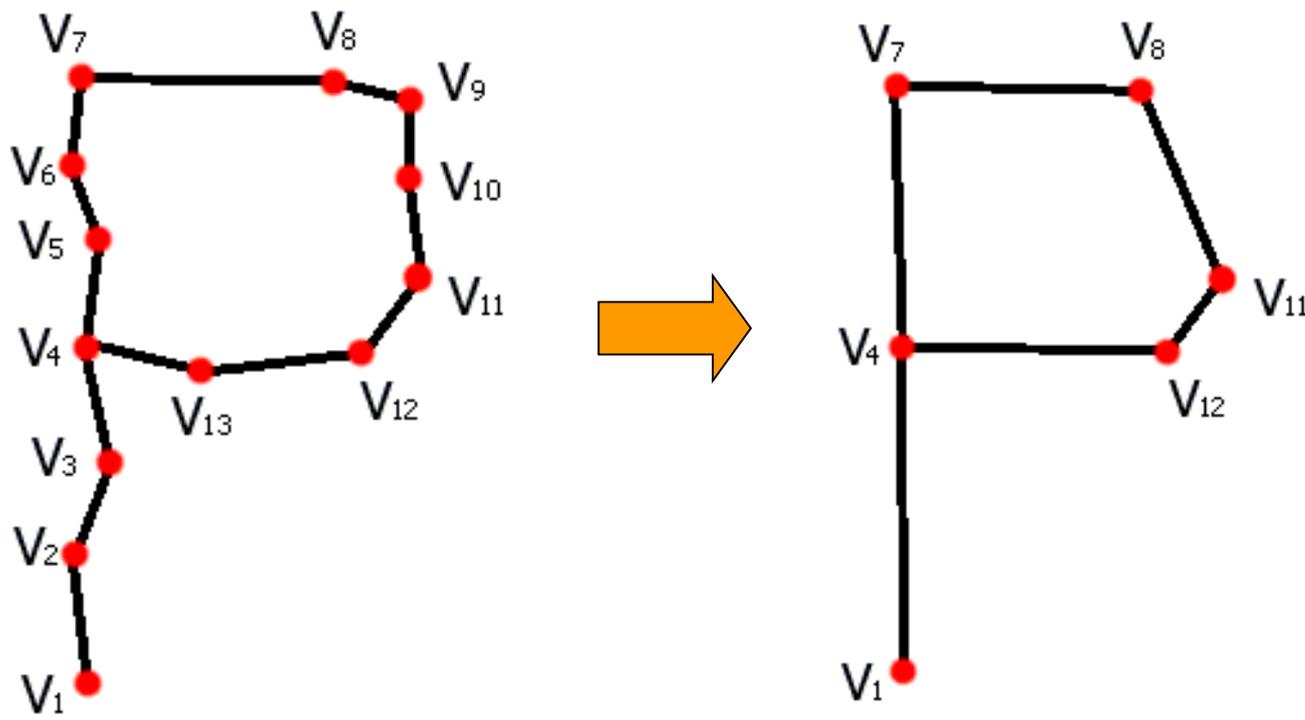
Необходимость ОПТИМИЗАЦИИ

- Полученный скелет изображения не является оптимальным. Это связано прежде всего с тем, что мы имеем дело с растровым изображением, а значит, изображение имеет искажения тем большие, чем меньше размер изображения в



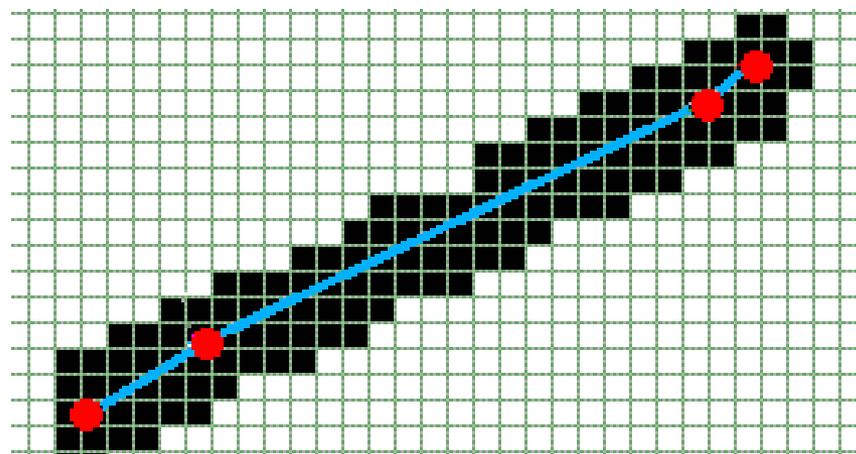
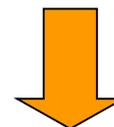
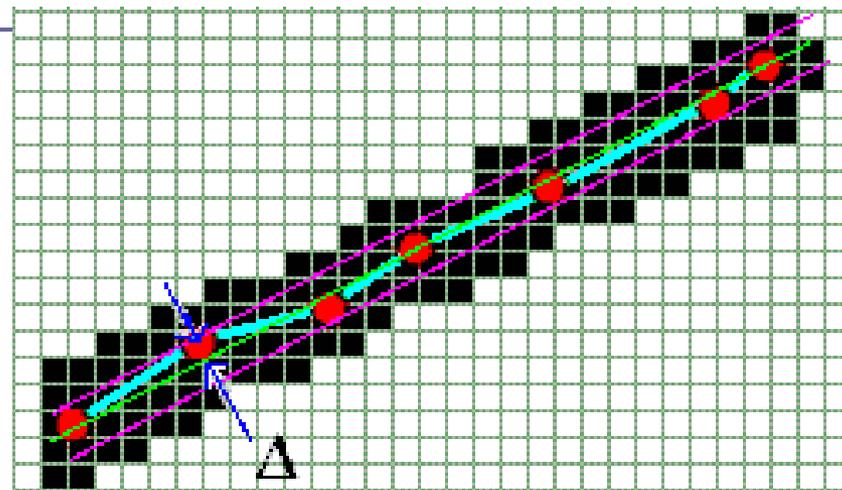
Оптимизация полученного скелета

- В получаемом скелете возможно представление одного отрезка некоторой последовательностью ребер.
- Избавиться от этого можно анализом последовательности ребер, оценивая отклонение получающейся линии от прямой.
- В случае, если отклонение находится в допустимых пределах, необходимо в скелете заменить соответствующую последовательность ребер на одно.



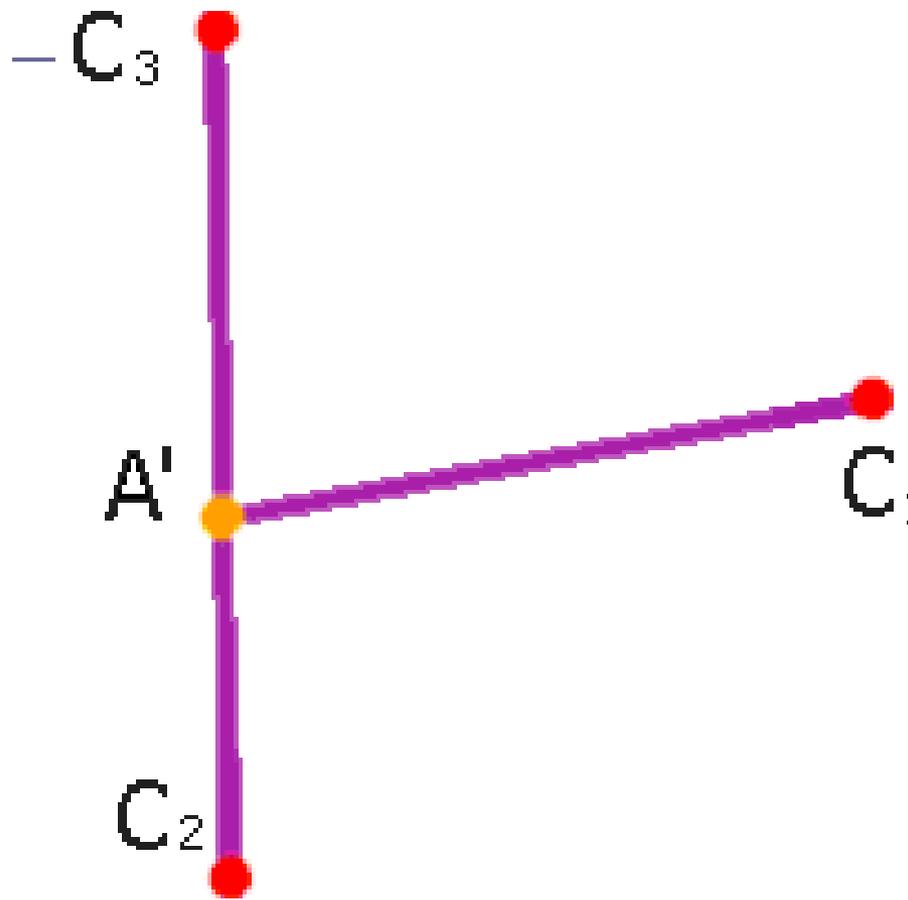
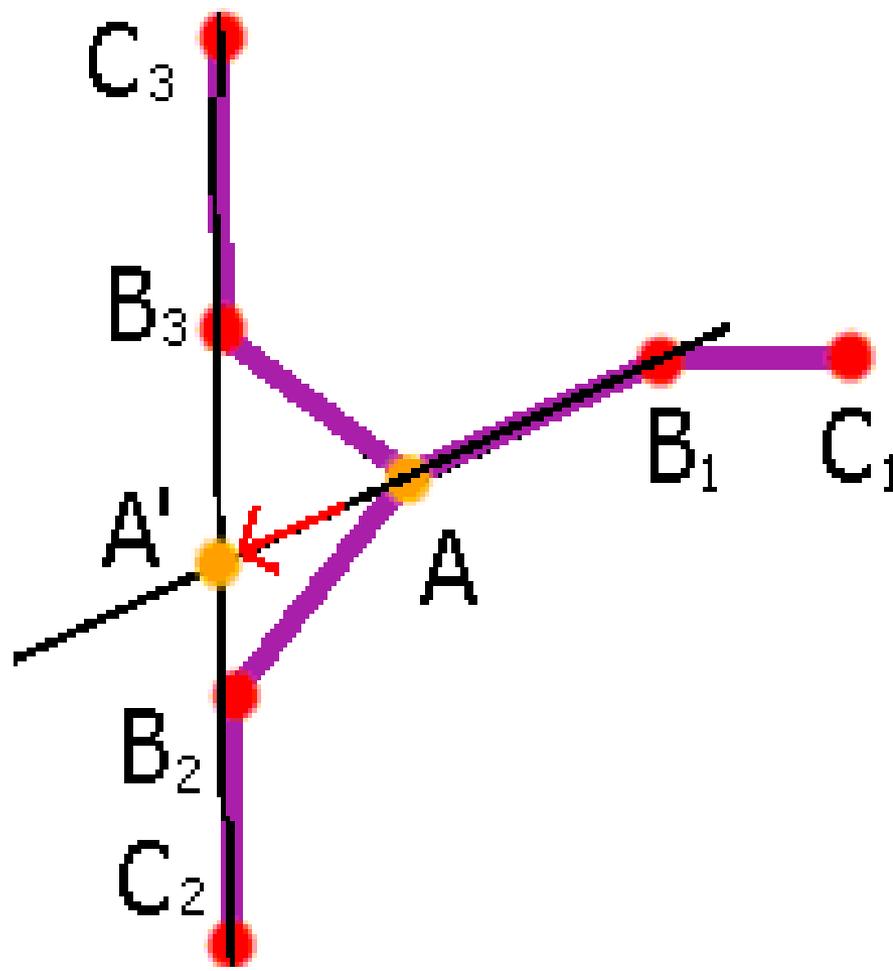
Оптимизация отрезков

- При оптимизации, точки, образующие последовательность ребер, должны отклоняться от прямой не больше, чем на заранее заданную величину, соизмеримую с шириной линии. От задаваемой величины зависит качество оптимизации: чем больше допустимое отклонение, тем меньше точек будет входить в



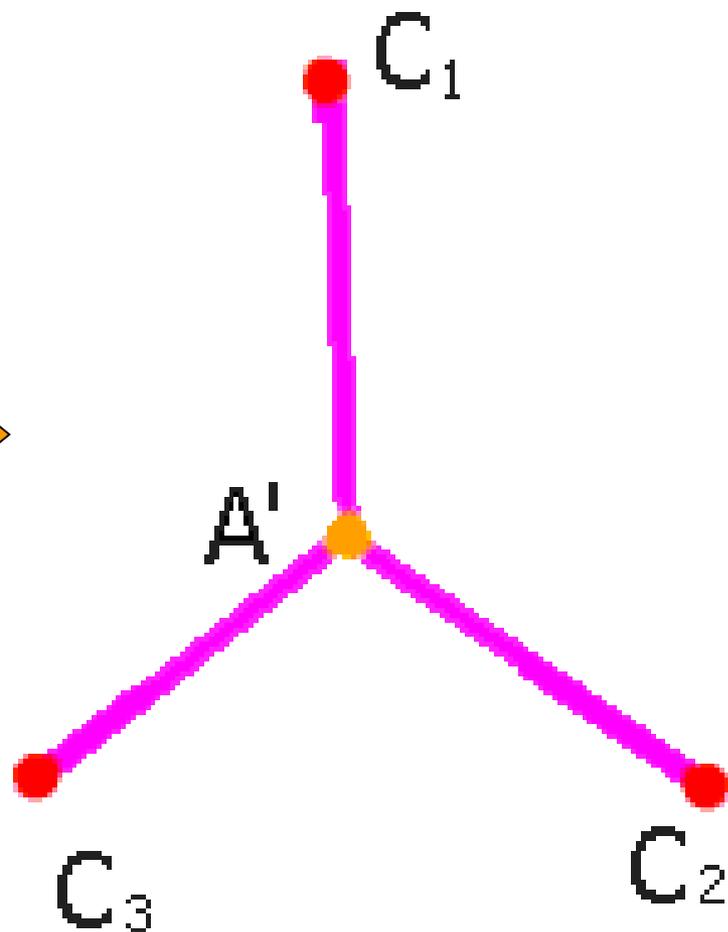
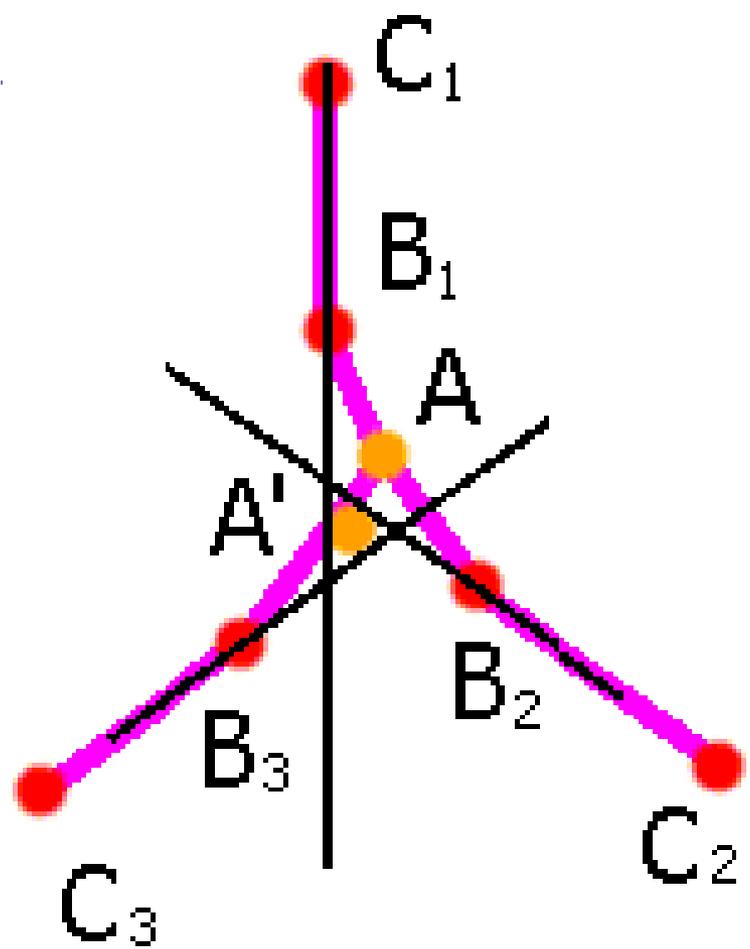
Оптимизация точки соединения отрезков

- Для оптимизации скелета просматриваются окрестности выделенных точек соединения отрезков.
- Искажения исправляются с помощью анализа прилежащих к выделенной точке (A) отрезков ($AB_1, B_1C_1, AB_2, B_2C_2, AB_3, B_3C_3$).
- Анализ заключается в поиске такой пары отрезков C_xV_x, V_yC_y из (B_1C_1, B_2C_2, B_3C_3) , что C_xV_x, V_yC_y максимально коррелируются прямой. Тогда необходимо точку A переместить в точку пересечения прямых C_xC_y и AC_2 , а затем удалить из графа точки B_1, B_2 и B_3



2-й вариант оптимизация точки соединения отрезков

- Другим вариантом искажения является случай соединения трех отрезков в одной точке.
- В этом случае невозможно нахождение пары отрезков коррелируемых прямой.
- Точка А должна быть перемещена в центр треугольника образуемого прямыми B_1C_1 , B_2C_2 и B_3C_3 . Затем точки B_1 , B_2 и B_3 необходимо удалить из графа



Сегментация



Разбиение изображения на неперекрывающиеся области (сегменты), покрывающие все изображение и однородные по некоторому признаку.

Уровни сегментации

Анализ высокого уровня:

- отделение находящихся на изображении объектов от фона (и друг от друга)

Анализ низкого уровня:

- разбиение на области «похожих» между собой пикселей

Типы сегментации

□ Автоматическая

- Сегментация производимая без взаимодействия с пользователем
 - Картинка на входе, регионы на выходе

□ Интерактивная

- Сегментация, управляемая пользователем, допускающая и/или требующая ввода дополнительной информации
 - Пример – «волшебная палочка» в Photoshop

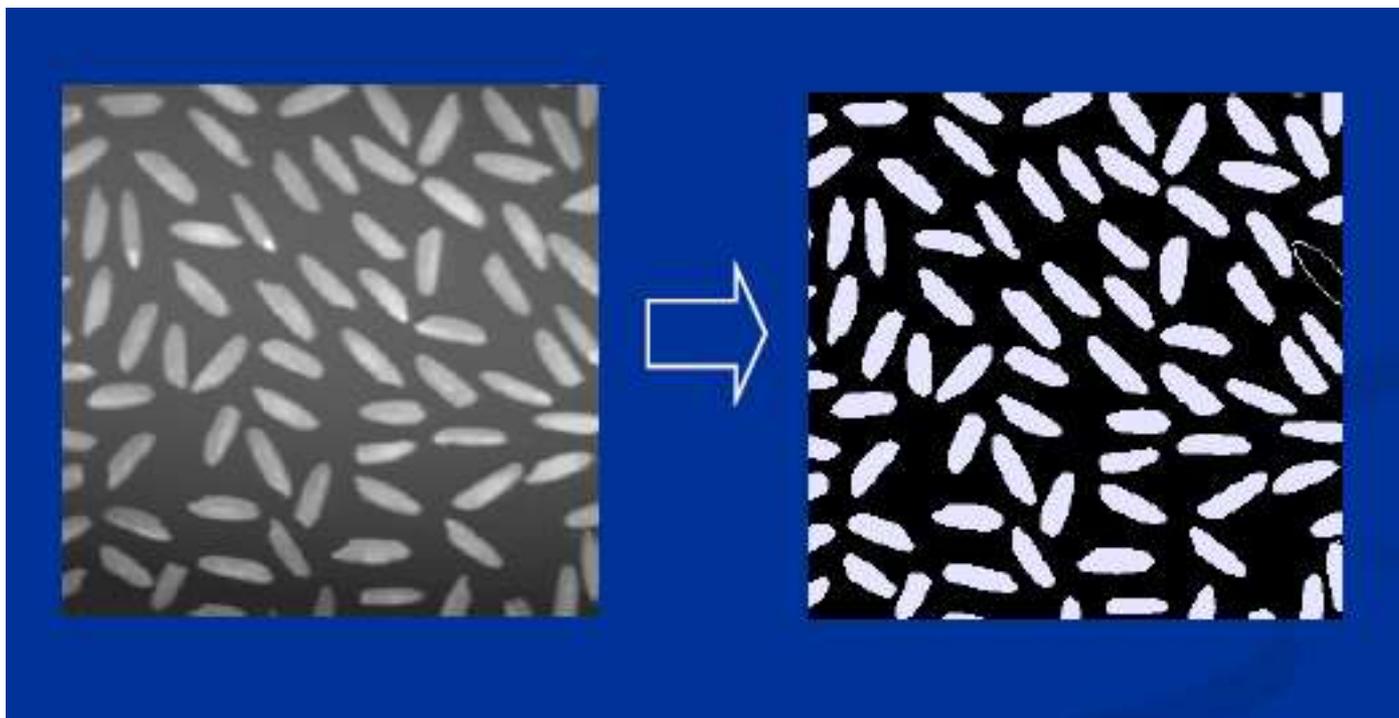
Применение сегментации

- Фото(видео)монтаж, композиция



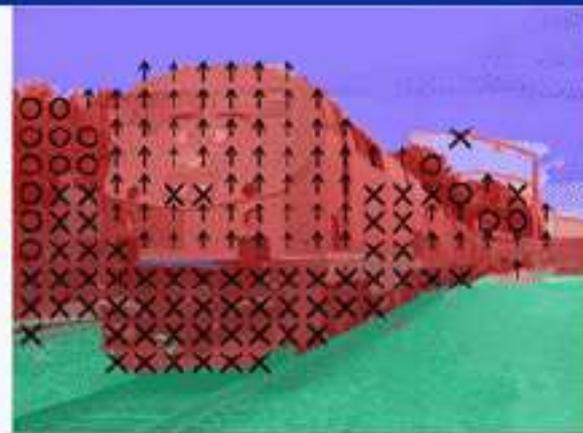
Применение сегментации

- Изменение параметров объектов



Применение сегментации

- ▣ Предобработка перед высокоуровневым анализом



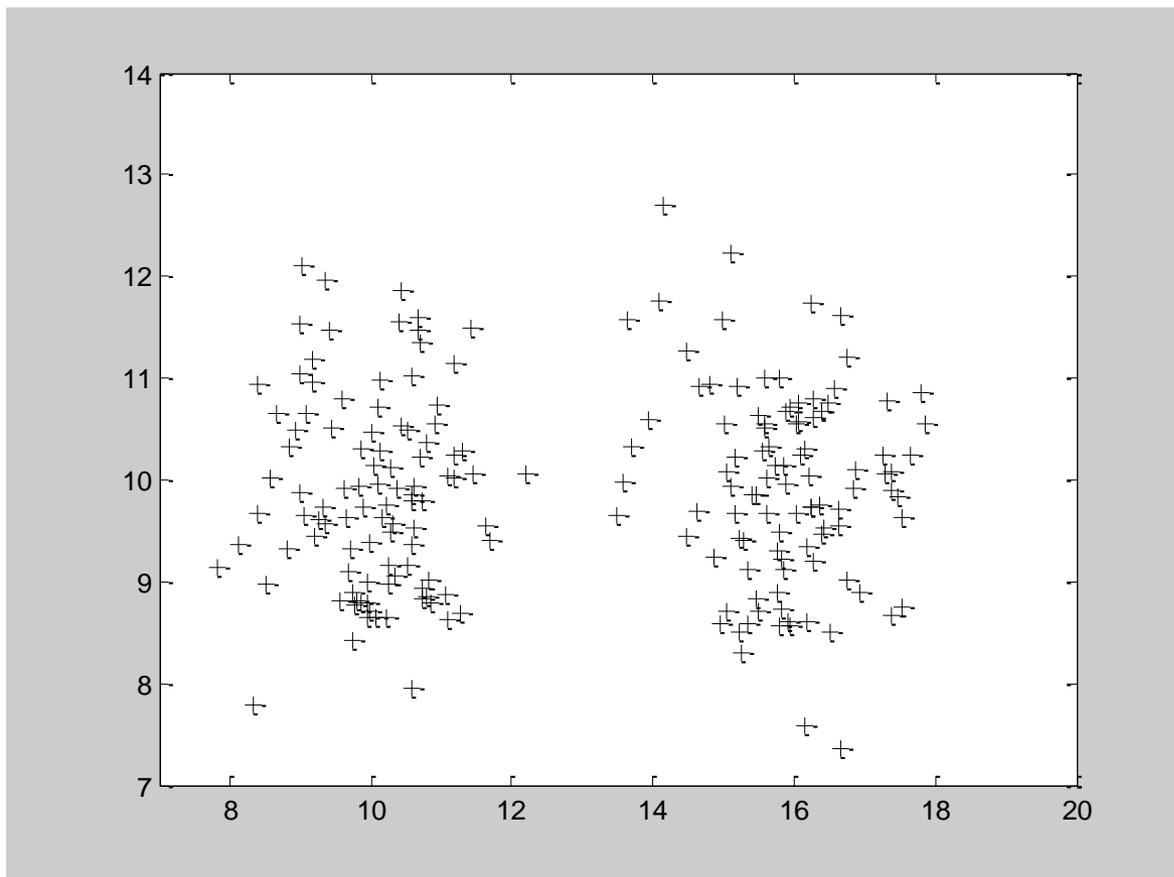
Автоматическая сегментация

- Как можно сформировать однородные области?
 - Отталкиваясь от неоднородности на границах
 - Пример – ищем резкие переходы яркости, берем их как границы областей
 - Отталкиваясь от однородности внутри Областей
 - Пример – объединяем в одну область пиксели, близкие по яркости

Методы основанные на кластеризации

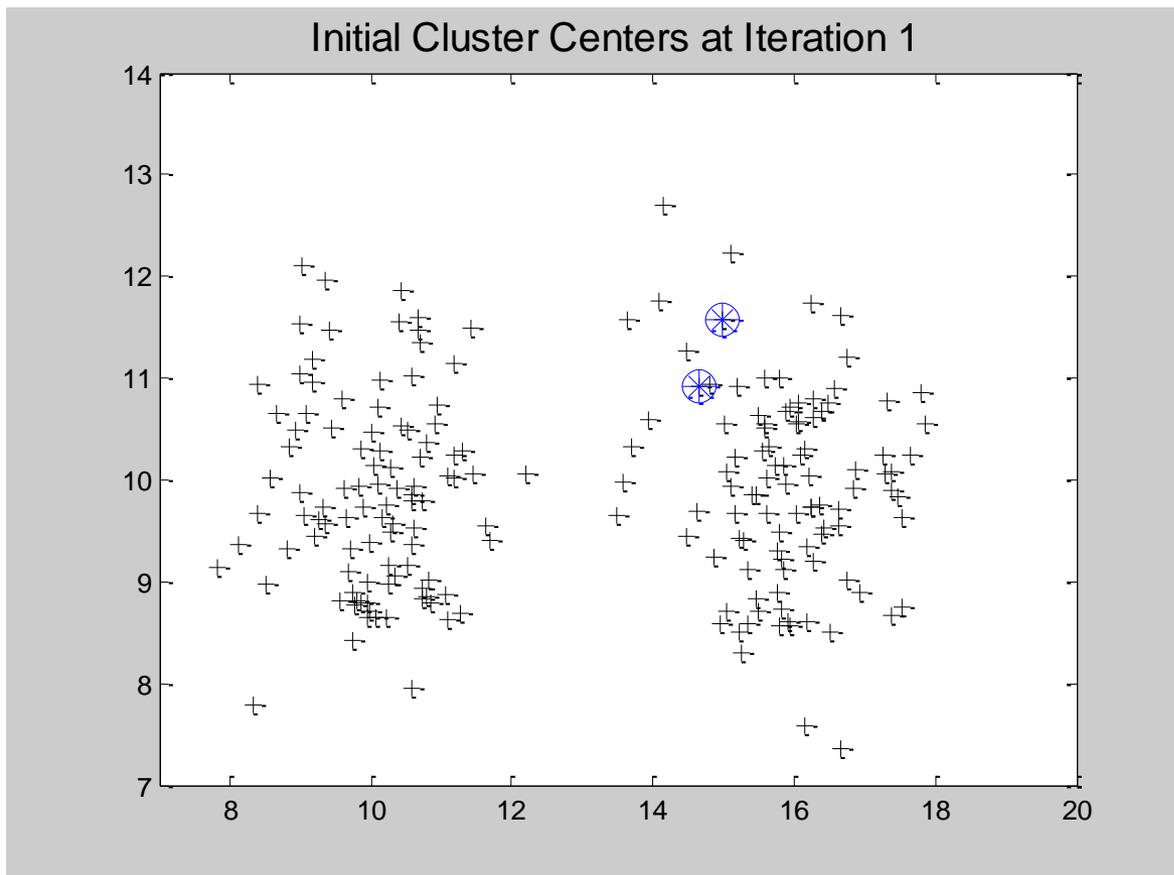
- k -средних — это итеративный метод, который используется, чтобы разделить изображение на K кластеров. Алгоритм :
 1. Выбрать K центров кластеров, случайно или на основании некоторой эвристики
 2. Поместить каждый пиксель изображения в кластер, центр которого ближе всего к этому пикселю
 3. Заново вычислить центры кластеров, усредняя все пиксели в кластере
 4. Повторять шаги 2 и 3 до сходимости (например, когда пиксели будут оставаться в том же кластере)

Пример кластеризации в 2D



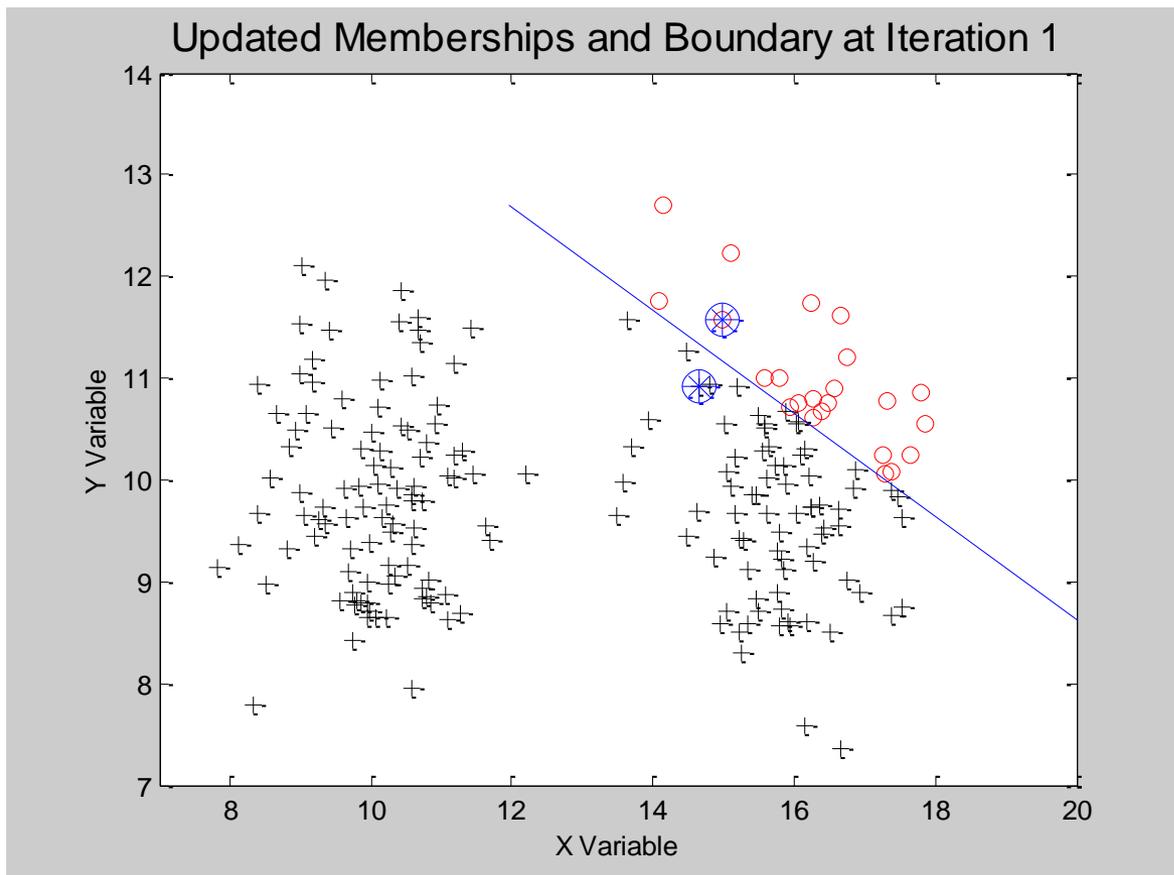
Исходные данные

Пример кластеризации в 2D



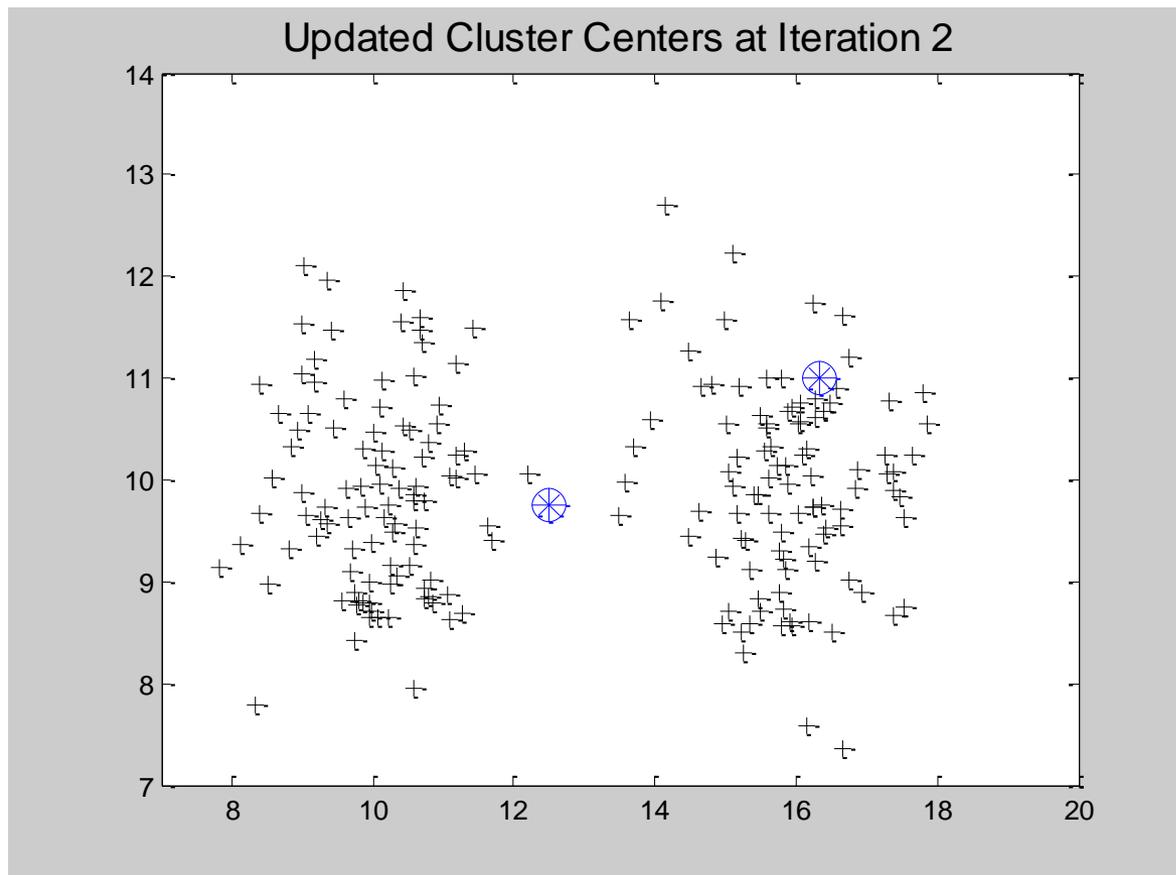
Случайная инициализация центров кластеров (шаг 1)

Пример кластеризации в 2D



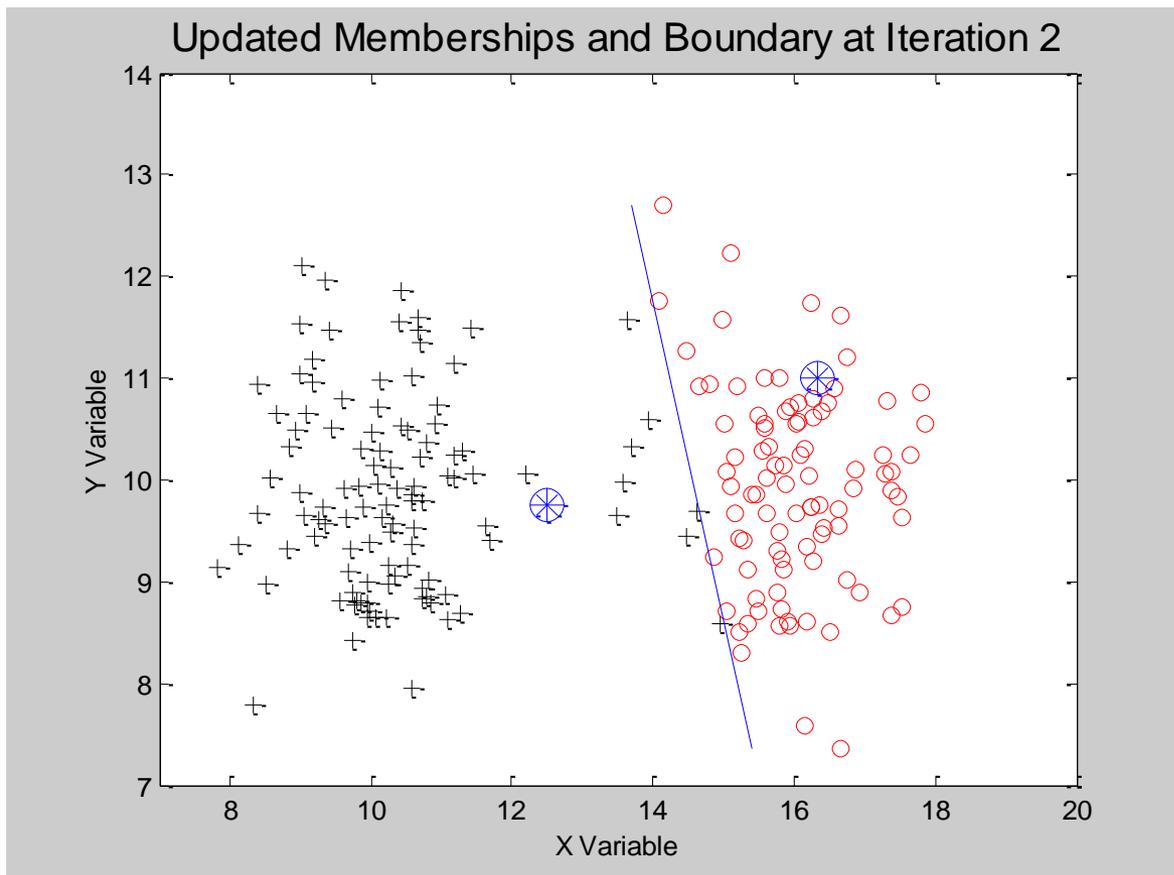
Кластеры после первой итерации (шаг 2)

Пример кластеризации в 2D



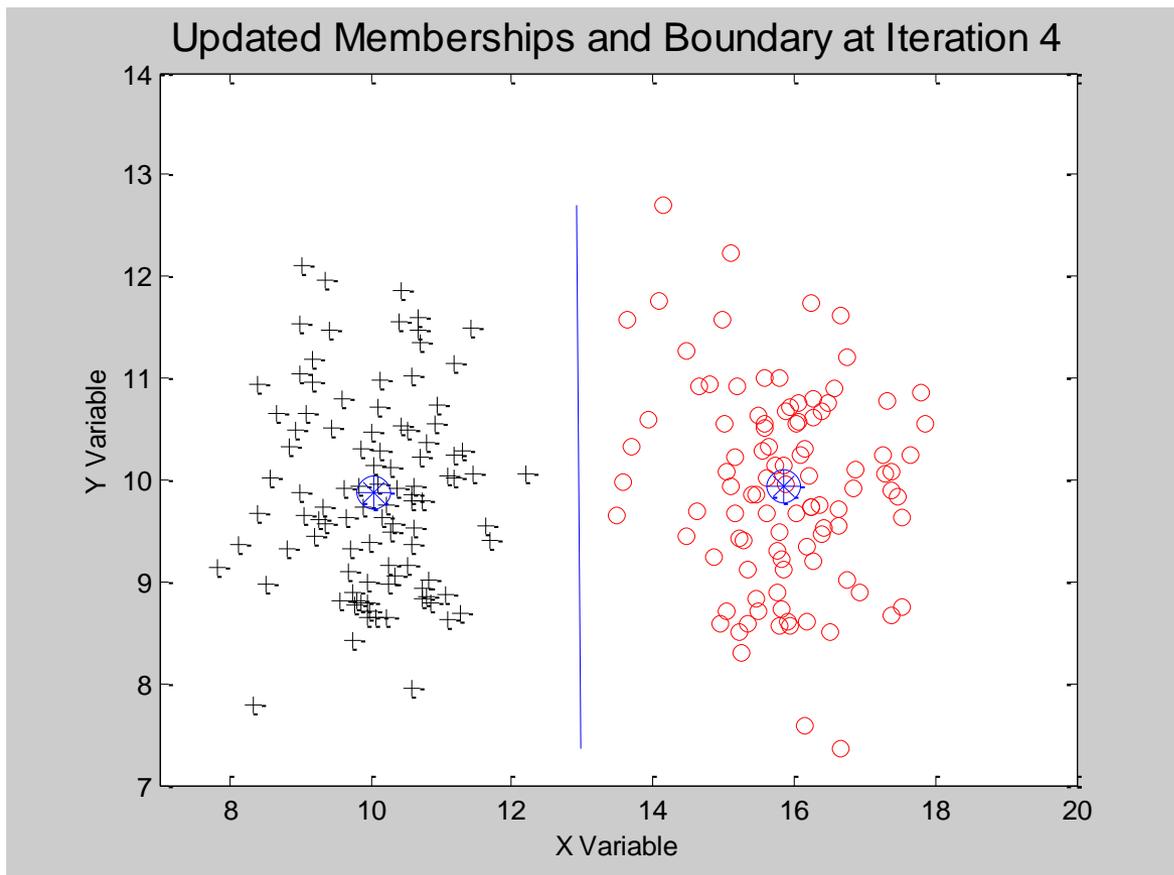
Пересчет центров кластеров после первой итерации (шаг 3)

Пример кластеризации в 2D



Кластеры после второй итерации (шаг 2)

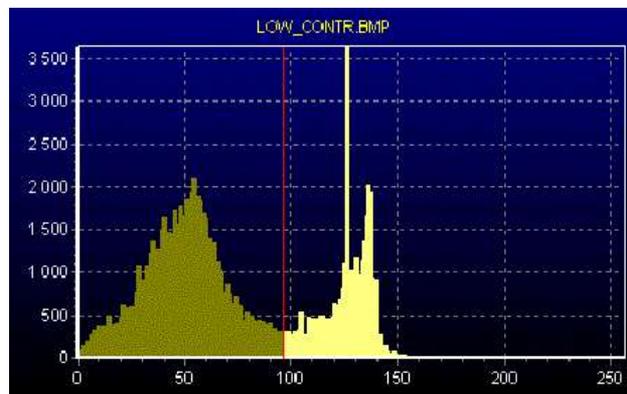
Пример кластеризации в 2D



Стабильная конфигурация после четвертой итерации

Применение k-средних для сегментации изображений по яркости

Рассматриваем *одномерное пространство яркостей пикселей* и производим в нем кластеризацию с помощью k-средних. Это дает автоматическое вычисление яркостных порогов.



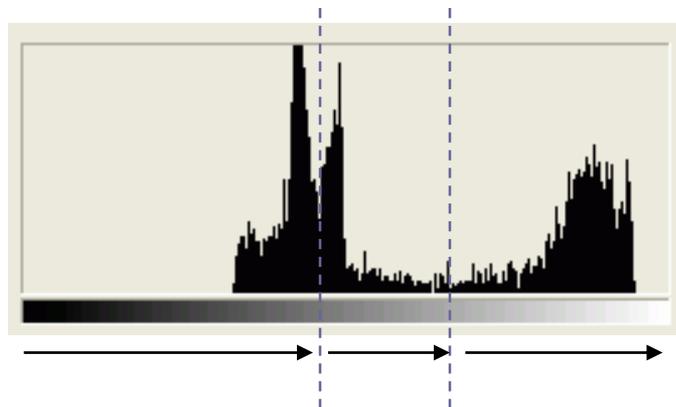
(Для получения бинарного изображения $k=2$)

Методы с использованием гистограммы

- Методы с использованием гистограммы очень эффективны, когда сравниваются с другими методами сегментации изображений, потому что они требуют только один проход по пикселям.
- В этом методе гистограмма вычисляется по всем пикселям изображения и её минимумы и максимумы используются, чтобы найти кластеры на изображении.

Сегментация с гистограммой

- Найти в гистограмме пики, разделить гистограмму по ним
- Для каждой части гистограммы найти связанные компоненты – это будут новые области



Алгоритм разрастания регионов

- Простая идея – начиная с некоторой “затравочной точки” обходить пиксели и объединять в области пока выполняется условие однородности (по цвету либо по яркости, либо по близости точек)

Сканируем изображение сверху вниз, слева направо:

1. if $|I(A) - Cl_{avg}(B)| > \delta$ and $|I(A) - Cl_{avg}(C)| > \delta$ -
создаем новую область, присоединяем к ней пиксел А

2. if $|I(A) - Cl_{avg}(B)| \leq \delta$ xor $|I(A) - Cl_{avg}(C)| \leq \delta$ -
добавить А к одной из областей

3. if $|I(A) - Cl_{avg}(B)| \leq \delta$ and $|I(A) - Cl_{avg}(C)| \leq \delta$:

1. $|Cl_{avg}(B) - Cl_{avg}(C)| \leq \delta$ -
сливаем области В и С.

2. $|Cl_{avg}(B) - Cl_{avg}(C)| > \delta$ -
добавляем пиксел А к тому классу, отклонение от
которого минимально.

$I(A)$ – яркость пиксела А

$Cl_{avg}(B)$ – средняя яркость области к которой принадлежит В

	С	
В	А	