Основы обработки изображений
Лекция 2.

- Понятие цветного, полутонового и бинарного изображения
- Алгоритмы бинаризации
- Логические операции над бинарными изображениями
- Базовая морфологическая обработка изображений

Цветные, полутоновые (монохромные) и бинарные (двоичные) изображения

В зависимости от задач, исследователям приходится работать с различными видами изображений, тем не менее многие часто путают монохромные и бинарные изображения.

Цветное изображение

В цветном изображении, содержится информация, позволяющая восстановить исходные цвета.

Чаще всего это достигается путём сохранения информации об интенсивности всех цветовых каналов.

Цветные изображения наиболее привычны и удобны для восприятия человеком, тем не менее в задачах автоматической обработки информация о цвете часто оказывается избыточной.

Полутоновое (монохромное) изображение

Полутоновое (монохромное) изображение содержит информацию об интенсивности света одной длины волны.

На практике полутоновые изображения чаще всего передаются в оттенках серого, но ничто не мешает представлять их в оттенках зелёного, красного или любого другого цвета.

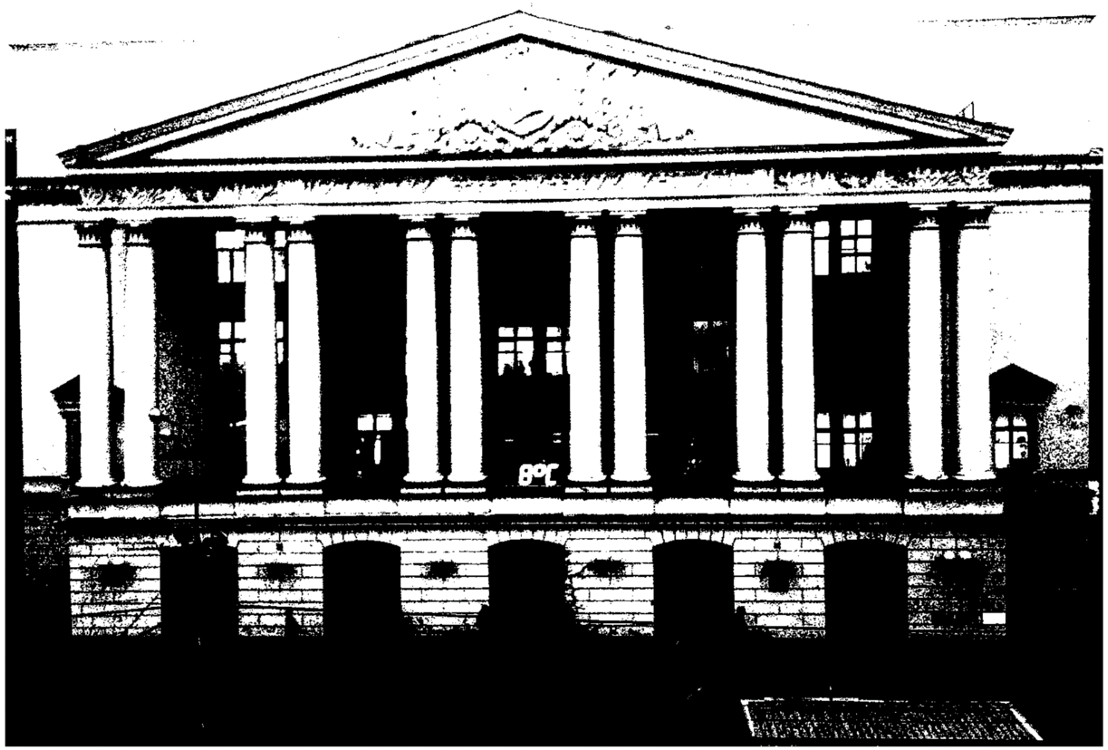
Монохромные изображения часто используются, как исходные данные для задач анализа изображений.

Бинарное (двоичное) изображение

В бинарных изображениях каждый пиксел может принимать только два значения. В зависимости от значения пикселы делят на пикселы фона (ПФ) и пикселы переднего плана (ППП).

Иногда бинарные изображения ошибочно называют чёрно-белыми. Надо понимать, что бинарным может быть и красно-чёрное изображение, при условии, что никаких других цветов не используется.

Несмотря на то, что бинарные изображения теряют значительную часть исходной информации, они очень часто используются на различных этапах обработки.



Алгоритмы бинаризации

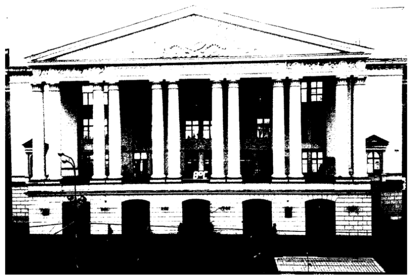
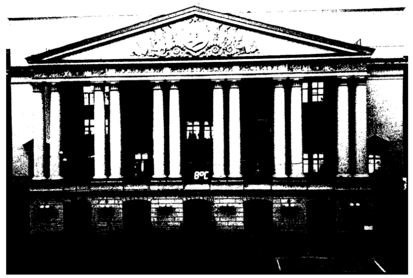
Многие задачи обработки изображений могут быть решены путём обработки бинарных изображений. Это порождает необходимость в разработке эффективных алгоритмов бинаризации.

Пороговая бинаризация

В простейшем случае используется бинаризация с одним порогом. Выбирается значение t, и все пикселы с яркостью не меньше t классифицируются как ППП, а все остальные пикселы классифицируются как ПФ. Такой способ бинаризации называется верхним пороговым. В случае нижней пороговой бинаризации объектами переднего плана будут считаться пикселы с яркостью не больше выбранного порога.

Чуть более сложным является алгоритм внутренней пороговой бинаризации. Выбирается два порога: верхний ($t_\text{в}$) и нижний ($t_\text{н}$). ППП будут считаться пикселы, яркость которых попадает в выбранный диапазон, а ПФ все остальные. Если используется обратный критерий, то говорят о внешней пороговой бинаризации.

Верхняя пороговая бинаризация	Нижняя пороговая бинаризация
$Img'[i,j] = \begin{cases} \text{ППП}, Img[i,j] \geq t \\ \text{ПФ}, Img[i,j] < t \end{cases}$	$Img'[i,j] = \begin{cases} \text{ППП}, Img[i,j] \leq t \\ \text{ПФ}, Img[i,j] > t \end{cases}$
Внутренняя пороговая бинаризация	**Внешняя пороговая бинаризация**
$Img'[i,j] = \begin{cases} \text{ПФ}, Img[i,j] < t_\text{н} \\ \text{ППП}, t_\text{н} \leq Img[i,j] \leq t_\text{в} \\ \text{ПФ}, Img[i,j] > t_\text{в} \end{cases}$	$Img'[i,j] = \begin{cases} \text{ППП}, Img[i,j] < t_\text{н} \\ \text{ПФ}, t_\text{н} \leq Img[i,j] \leq t_\text{в} \\ \text{ППП}, Img[i,j] > t_\text{в} \end{cases}$

Изображение	Порог 100	Порог 125
		

Неполная пороговая бинаризация

В некоторых случаях необходимо избавиться от определённого диапазона яркостей, сохранив при этом информацию о яркости остальных пикселей. В таких случаях используется неполная пороговая бинаризация.

Это преобразование зачастую упрощает дальнейший анализ изображения (например, сегментацию), т.к. многие несущественные детали становятся ПФ, а важные детали полностью сохраняют информацию об интенсивности.

Верхняя неполная пороговая бинаризация	**Нижняя неполная пороговая бинаризация**
$Img'[i,j] = \begin{cases} Img[i,j], Img[i,j] \geq t \\ \text{ПФ}, Img[i,j] < t \end{cases}$	$Img'[i,j] = \begin{cases} Img[i,j], Img[i,j] \leq t \\ \text{ПФ}, Img[i,j] > t \end{cases}$
Внутренняя неполная пороговая бинаризация	**Внешняя неполная пороговая бинаризация**
$Img'[i,j] = \begin{cases} \text{ПФ}, Img[i,j] < t_{\text{н}} \\ Img[i,j], t_{\text{н}} \leq Img[i,j] \leq t_{\text{в}} \\ \text{ПФ}, Img[i,j] > t_{\text{в}} \end{cases}$	$Img'[i,j] = \begin{cases} Img[i,j], Img[i,j] < t_{\text{н}} \\ \text{ПФ}, t_{\text{н}} \leq Img[i,j] \leq t_{\text{в}} \\ Img[i,j], Img[i,j] > t_{\text{в}} \end{cases}$

Изображение

Порог 100

Порог 125

Бинаризация Бернсена

В методе Бернсена для каждого пиксела $Img[i,j]$ выбирается свой порог, зависящий от яркости близлежащих пикселов.

В качестве окрестности чаще всего рассматривается квадрат с центром в обрабатываемом пикселе. Порогом для исследуемого пиксела выбирается значение $I = \frac{I_{max} - I_{min}}{2}$, где I_{max} – максимальный уровень яркости в окрестности, а I_{min} – минимальный уровень яркости в окрестности.

При этом, если уровень контраста (значение $I_{max} - I_{min}$) превышает некоторый заранее установленный порог, бинаризация происходит не по порогу, а пиксел принимает заранее определённое для таких ситуаций значение

Логические и теоретико-множественные операции над бинарными изображениями

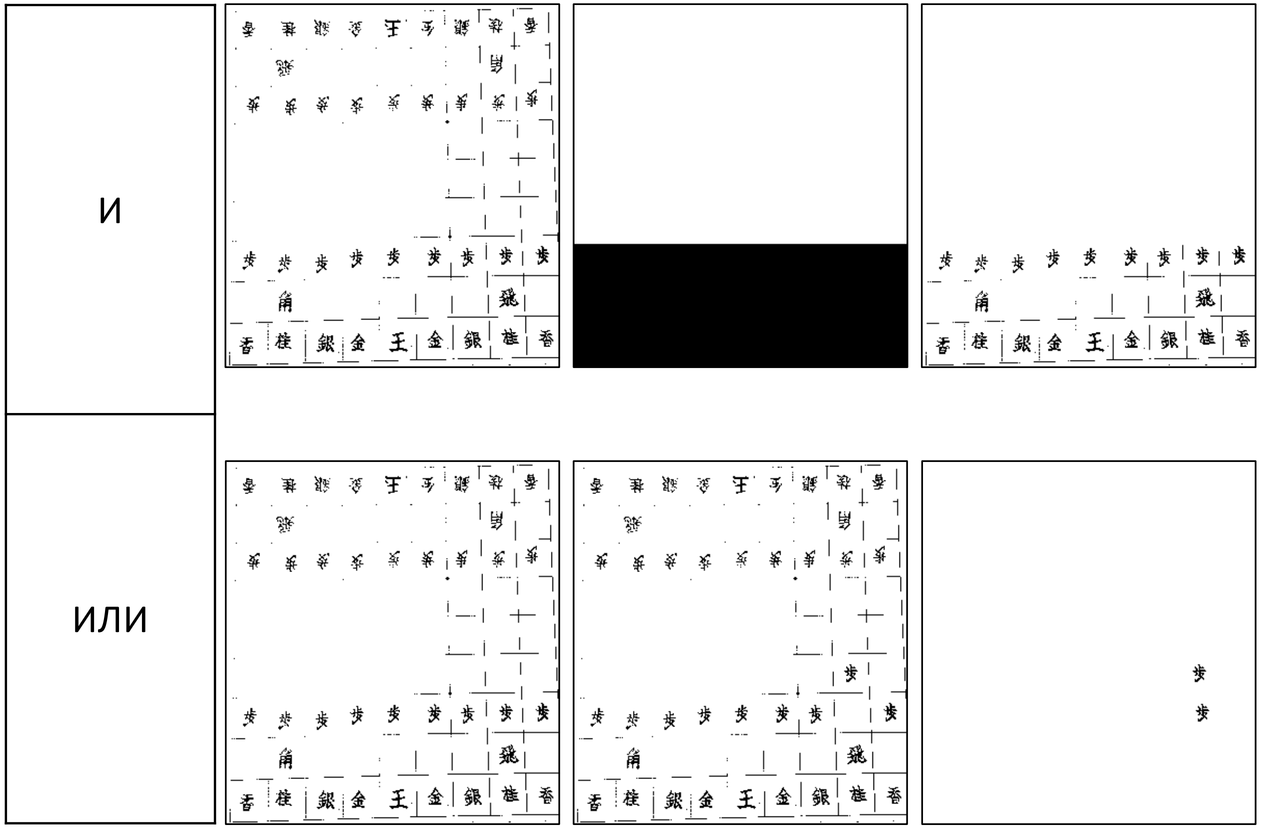
Простота бинарных изображений позволяет легко применять к ним широко используемые математические методы обработки

Логические операции над бинарными изображениями

Особенностью бинарных изображений является возможность применения к ним операций классической логики:

- Логическое И;
- Логическое ИЛИ;
- Отрицание;
- Другие логические операции и их комбинации.

Все логические операции (за исключением операции отрицания) выполняются над двумя изображениями, одно из которых является исходным, а второе в большинстве случаев – или специально сгенерированной маской, или обработанным исходным изображением. Применение логических операций к изображению позволяет упростить его дальнейшую обработку. Например, используя операцию логического И, можно выделить часть изображения, предназначенную для дальнейшего анализа.



Теоретико-множественные операции

Изображение можно рассматривать, как подмножество A двумерного целочисленного пространства Z^2. Если какой-то элемент $a = (a_1, a_2)$ является элементом множества A, то это записывается так: $a \in A$, в противном случае так: $a \notin A$. Если в множество не содержится ни одного элемента, оно называется пустым и обозначается $\emptyset$.

Один из способов задания множества – перечисление элементов, входящих в него. Например, запись $A = \{1,2,3\}$ задаёт множество из трёх элементов $1,2,3$, а запись $B = \{b | b = \pm a, a \in A\}$ задаёт множество, состоящее из шести элементов: $\pm 1, \pm 2, \pm 3$. Иногда такую запись называют конструктором множества. Также заметим, что все элементы A входят в B, поэтому множество A называется подмножеством множества B и обозначается $A \subseteq B$.

Операции над множествами

Объединением множеств A и B называется множество $C = A \cup B$, содержащие элементы, принадлежащие либо множеству A, либо множеству B, либо обоим множествам одновременно.

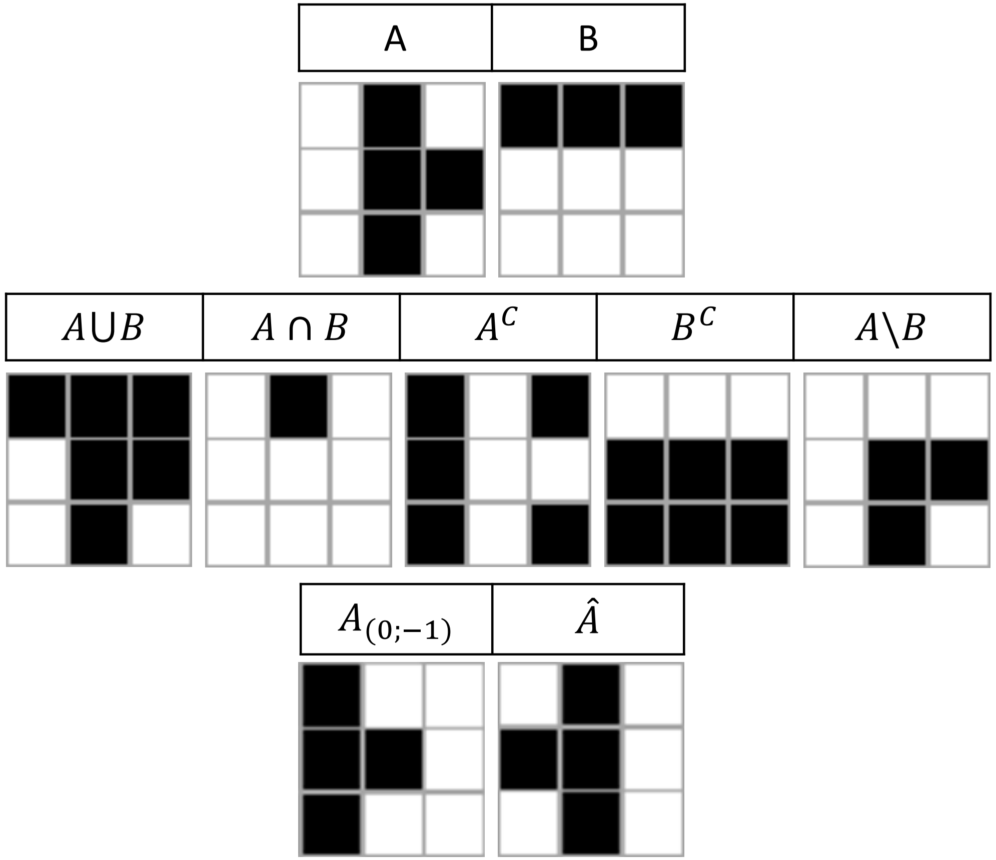
Пересечением множеств A и B называется множество $C = A \cap B$, содержащие только элементы, принадлежащие обоим множествам одновременно.

Дополнением множества называют элементы, не входящие в множество: A^C.

Разностью множеств A и B называется множество $C = A \backslash B$, содержащие элементы A, которых нет в B.

Операция параллельного переноса определяется следующим образом: $(A)_t = \{e | e = a + t, a \in A\}$. Вектор переноса t чаще всего задаётся в виде пары чисел $(\Delta r, \Delta c)$.

Операция центрального отражения определяется следующим образом: $\hat{A} = \{e | e = -a, a \in A\}$. Эта операция выполняется относительно начала координат.



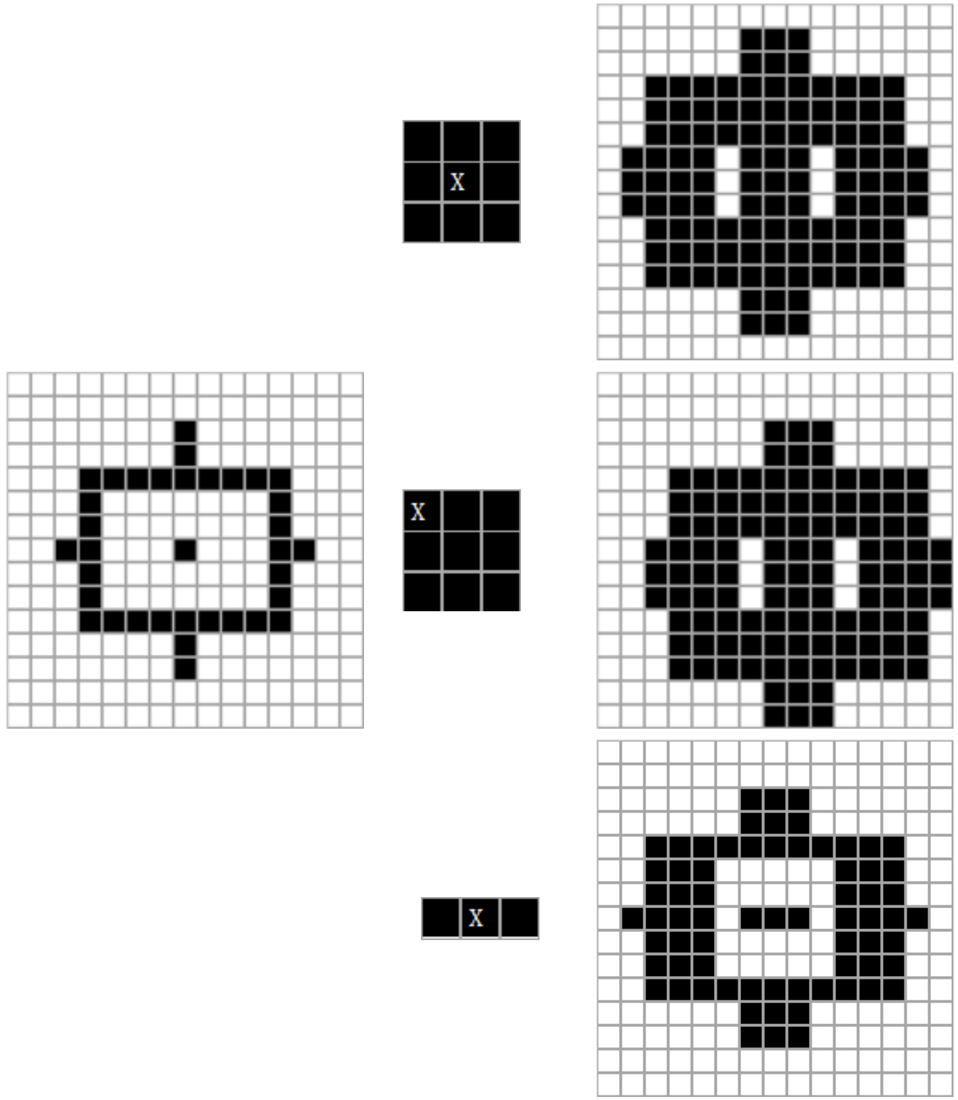
Математическая морфология

Термином «морфология» обычно обозначают области, изучающие формы и структуры объектов. Математическая морфология изучает структуры и формы множеств – изображений.

Наращивание (дилатация)

Дилатацию изображения B структурирующим элементом S можно описать следующим выражением: $B \oplus S = \{t | (\hat{S})_t \cap B \neq \emptyset\}$.

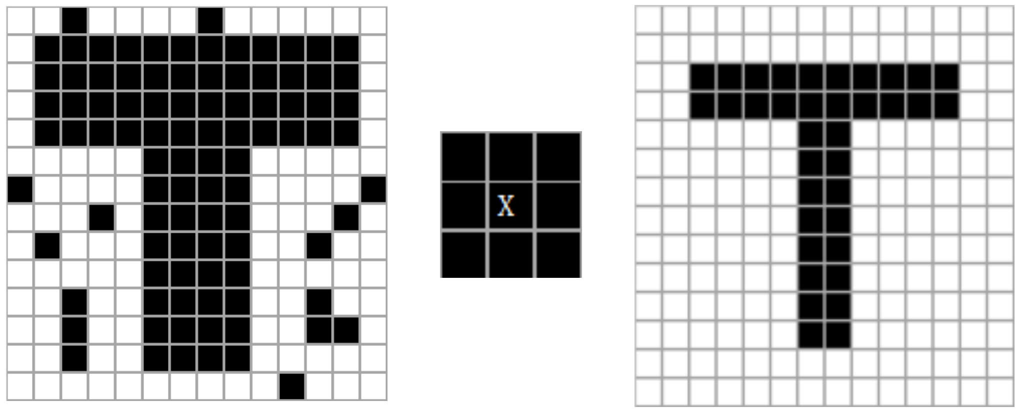
Для каждого единичного пиксела исходного изображения выполняется перенос структурирующего элемента так, чтобы его начало координат совпало с рассматриваемым единичным пикселом, затем выполняется логическое сложение структурирующего элемента и соответствующих пикселов изображения. Результат сложения записывается в итоговое изображение, изначально инициализированное нулевыми пикселами.



Эрозия

Эрозия изображения B структурирующим элементом S описывается следующим выражением: $B \ominus S = \{t | (S)_t \subseteq B\}$.

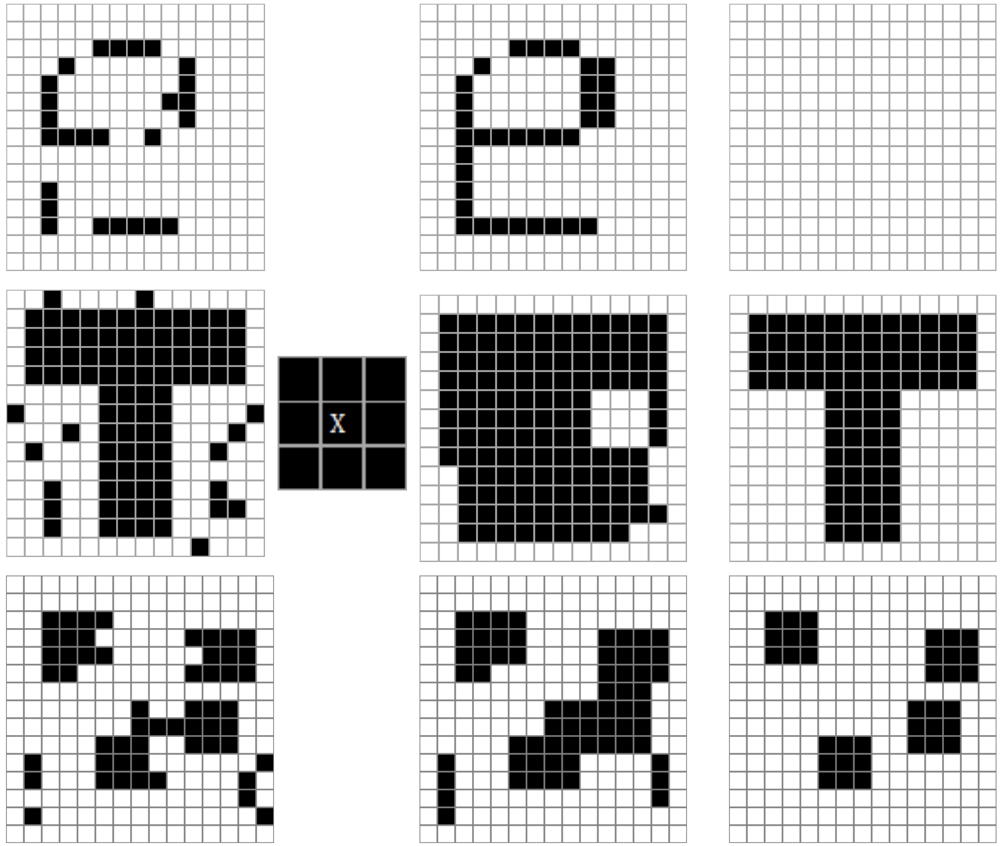
Как и в случае с наращиванием, выполняется перенос структурирующего элемента, таким образом, чтобы начало координат структурирующего элемента совпало с исследуемым пикселом изображения, но в отличие от дилатации совмещение происходит со всеми пикселами изображения, а не только с единичными. Если после какого-либо переноса каждый единичный пиксел структурирующего элемента совпал с единичным пикселом исходного изображения, то в соответствующий пиксел итогового изображения записывается значение, соответствующее ППП.



Замыкание и размыкание

Замыканием изображения B структурирующим элементом S называется итоговое изображение $B \cdot S = (B \oplus S) \ominus S$. Аналогично размыканием изображения B структурирующим элементом S называется итоговое изображение $B \circ S = (B \ominus S) \oplus S$. Обе эти операции сглаживают контуры изображения, но делают это по-разному.

Замыкание приводит к закрытию небольших разрывов и отверстий, а размыкание приводит к разрыву узких перешейков и удалению небольших выступов.



ВОПРОСЫ

?