1. **Build up linked list**

In the memory, linked lists, one kind of data structure, are used for each individual pattern. Doubly linked list is chosen to make algorithm faster than traditional linked list. Routines like building lists and deleting nodes are implemented. There is a head and end node in each list. Head and end node contain no data, just facilitate deleting node. The linked list is superior to array, because when each pixel is processed, the linked list becomes shorter and shorter. This makes the locating of a pixel in the list more efficient. There is no time wasted on bypassing those pixels have been processed for array scenario.

The following is the paradigm of linked lists.

![Figure 1. Linked List](image1)

![Figure 2. Doubly Linked List](image2)
Doubly linked list is faster than traditional linked list, because it can search from two ends while traditional linked list can search from head.

While for array, its size is constant, for each pixel, the algorithm needs to go through the array all the time. It is really time consuming for huge continental image aggregation, it takes about 3 months to get aggregation done by estimation. The array model is as following.

<table>
<thead>
<tr>
<th>Item 1</th>
<th>Item 2</th>
<th>Item 3</th>
<th>Item 4</th>
<th>Item 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pix1</td>
<td>Pix2</td>
<td>Pix3</td>
<td>Pix4</td>
<td>Pix5</td>
</tr>
<tr>
<td>Location</td>
<td>Location</td>
<td>Location</td>
<td>Location</td>
<td>Location</td>
</tr>
<tr>
<td>Biomes</td>
<td>Biomes</td>
<td>Biomes</td>
<td>Biomes</td>
<td>Biomes</td>
</tr>
</tbody>
</table>

After processing each pixel, the array size is always the same. Algorithm needs to check pixel by pixel which has not been processed. It will take too much time. For the huge continental biome classification image used in this research, array is infeasible.

**2. Navigating doubly link list**

![Diagram of doubly linked list](image)

Figure 3. Relationship between linked pixel info list to linked block info list
Figure 3 shows that there are two doubly linked lists, linked pixel info list and linked block info list. It is just an illustration. The block means a linked block info list node with the number of pixels it represents to be aggregated. In the program implemented, the regular block represents 1000 pixel, with the exception of the last block having pixel number less than 1000. For each linked list, there are a head and an end node, which are for the convenience of list operation, like node removal. Each node in linked pixel info list has the information of pixel, i.e., class type of the four subpixels, a pixel index, its location (row, column) in the aggregated image. Each block node consists of the pixel node address in memory, the block size, and a block index. For each pixel spatial pattern, there is a pair of these doubly linked lists, a doubly pixel info list and a doubly linked block info list. There are 6 pairs of lists for the first 6 patterns respectively. Both linked pixel info list and linked block info list above are dynamic. After a pixel is aggregated, the pixel will be removed from the linked pixel info list, and its corresponding block info list block size will be deducted by 1. When block size reaches 0, the block will be removed from the linked block info list.

Introducing block info list expedite locating of a pixel within the linked pixel info list about 1705 times than without it for pattern 1 which is the biggest heterogenous pattern. Doubly linked lists and the block linked list locating pixel are applied in this research.