Got it. If you want to keep both previews, you can modify the code accordingly. Here's the updated version:

```
function handleImageInputChange() {
  const fileInput = document.getElementById('imageInput');
  const imageUrlInput = document.getElementById('imageUrl');
  const uploadPreview = document.getElementById('backgroundPreview');
  const urlPreview = document.getElementById('urlBackgroundPreview');

  if (fileInput.files && fileInput.files[0]) {
    const reader = new FileReader();

    reader.onload = function (e) {
      // Create a new image element
      const img = new Image();

      img.onload = function () {
        // Create a canvas element
        const canvas = document.createElement('canvas');
        const ctx = canvas.getContext('2d');

        // Resize the image to a maximum size of 1000x1000
        const maxDimension = 1000;
        const scaleFactor = Math.min(maxDimension / img.width,
maxDimension / img.height);
        canvas.width = img.width * scaleFactor;
        canvas.height = img.height * scaleFactor;

        // Draw the image onto the canvas
        ctx.drawImage(img, 0, 0, canvas.width, canvas.height);

        // Set the uploadPreview.src to the data URL of the resized image
        uploadPreview.src = canvas.toDataURL();

        // Display the uploadPreview
        uploadPreview.style.display = 'block';

        // Set the urlPreview.src to the empty string
        urlPreview.src = '';

        // Display the urlPreview
        urlPreview.style.display = 'none';
      };

      // Set the source of the image to the data URL
      img.src = e.target.result;
    };

    reader.readAsDataURL(fileInput.files[0]);
  }
}
```

In this version, the `urlPreview.src` is set to an empty string, and its display is set to 'none' to hide it. This way, both previews are retained, and only the `uploadPreview` is displayed when a new image is selected.