```
function [ warpedImage ] = warpImage( rgbImage, H )
    totalImageSize = size(rgbImage);
    imageHeight = totalImageSize(1);
    imageWidth = totalImageSize(2);

    dummyVal = -100000;
    minX = dummyVal;
    minY = dummyVal;
    maxX = dummyVal;
    maxY = dummyVal;

    for x = 1:imageHeight
        for y = 1:imageWidth
            newPoints = H * [x; y; 1];
            newX = round(newPoints(1));
            newY = round(newPoints(2));
            if (newX < minX || minX == dummyVal)
                minX = newX;
            end
            if (newY < minY || minY == dummyVal)
                minY = newY;
            end
            if (newX > maxX || maxX == dummyVal)
                maxX = newX;
            end
            if (newY > maxY || maxY == dummyVal)
                maxY = newY;
            end
        end
    end

    if (minX < 0)
       offsetX = minX * -1 + 1;
       offsetY = minY * -1 + 1;
    else
       offsetX = minX;
       offsetY = minY;
    end

    warpedImageSize = [(maxX - minX + 1) (maxY - minY + 1) 3];
    warpedImage = zeros(warpedImageSize);
    warpedImageHeight = warpedImageSize(1);
    warpedImageWidth = warpedImageSize(2);

    for x = 1:warpedImageHeight
        for y = 1:warpedImageWidth
            oldPoints = zeros(2, 1);
            if (minX < 0)
                oldPoints = H \ [x - offsetX; y - offsetY; 1];
            else
                oldPoints = H \ [x + offsetX; y + offsetY; 1];
            end
```

```
            oldX = round(oldPoints(1));
            oldY = round(oldPoints(2));

            if (oldX >= 1 && oldX <= imageHeight && oldY >=1 && oldY
 <= imageWidth)
                warpedImage(x, y, :) = rgbImage(oldX, oldY, :);
            end
        end
    end
end
```

Published with MATLAB® R2017a