```
<!DOCTYPE html>
<html lang="nl"><head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Fractale Vierkanten Animatie</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            margin: 0;
            background-color: lightblue;
        }
        .container {
            position: relative;
            width: 890px;
            height: 890px;
            display: flex;
            justify-content: center;
            align-items: center;
        }
        .black-box {
            position: absolute;
            width: 890px;
            height: 890px;
            background-color: black;
        }
        canvas {
            position: absolute;
            width: 890px;
            height: 890px;
        }
    </style>
</head>
<body>
    <div class="container">
        <div class="black-box"></div>
        <canvas id="myCanvas" width="890" height="890"></canvas>
    </div>

    <script>
        const canvas = document.getElementById("myCanvas");
        const ctx = canvas.getContext("2d");

        let time = 0;
        let speed = 0.002;
        let baseSize = 400;
        let scaleFactor = 0.63;
        let levels = 7; // Aantal fractale niveaus
        let rotationSpeed = 0.01;

        function drawFractal(x, y, size, level, rotation) {
            if (level > levels) return;

            ctx.save();
            ctx.translate(x, y);
            ctx.rotate(rotation);
            ctx.translate(-x, -y);

            // Bereken nieuwe lijnbreedte tussen 0.2 en 0.5
            ctx.lineWidth = 0.2 + (0.3 * (levels - level) / levels);

            let halfSize = size / 2;
            let x1 = x - halfSize;
            let y1 = y - halfSize;
```

```
            let x2 = x + halfSize;
            let y2 = y + halfSize;

            let colors = ["rgb(35, 95, 255)", "rgb(230, 120, 180)", "rgb(75, 210, 220)",
                          "rgb(190, 60, 255)", "rgb(230, 30, 255)", "rgb(250, 30, 255)",
                          "rgb(190, 230, 155)", "rgb(255, 255, 115)", "rgb(255, 80, 95)"
                          ];

            ctx.strokeStyle = colors[level % colors.length];
            ctx.beginPath();
            ctx.rect(x1, y1, size, size);
            ctx.stroke();

            // Recursieve fractale vierkanten
            let newSize = size * scaleFactor;
            let angleOffset = Math.sin(time * 2 * Math.PI) * 0.1;
            drawFractal(x1, y1, newSize, level + 1, rotation + angleOffset);
            drawFractal(x2, y1, newSize, level + 1, rotation - angleOffset);
            drawFractal(x1, y2, newSize, level + 1, rotation + angleOffset);
            drawFractal(x2, y2, newSize, level + 1, rotation - angleOffset);

            ctx.restore();
        }

        function animate() {
            time += speed;
            ctx.clearRect(0, 0, canvas.width, canvas.height);
            drawFractal(canvas.width / 2, canvas.height / 2, baseSize, 0, time *
            rotationSpeed);
            requestAnimationFrame(animate);
        }

        animate();
    </script>


</body></html>
```