3个动画：

* 下落
* 挤压和拉伸
  + X变小
  + Y变大
  + 高度变小
  + 跨度变大
* 上升

ShapeHolder newBall = addBall(event.getX(), event.getY());

// Bouncing animation with squash and stretch

**float** startY = newBall.getY();

**float** endY = getHeight() - 50f;

**float** h = (**float**)getHeight();

**float** eventY = event.getY();

**int** duration = (**int**)(500 \* ((h - eventY)/h));

//下落动画

ValueAnimator bounceAnim = ObjectAnimator.*ofFloat*(newBall, "y",

startY, endY);

bounceAnim.setDuration(duration);

bounceAnim.setInterpolator(**new** AccelerateInterpolator());

//挤压与拉伸，x变小

ValueAnimator squashAnim1 = ObjectAnimator.*ofFloat*(newBall,

"x", newBall.getX(),newBall.getX() - 25f);

squashAnim1.setDuration(duration/4);

squashAnim1.setRepeatCount(1);

squashAnim1.setRepeatMode(ValueAnimator.*REVERSE*);

squashAnim1.setInterpolator(**new** DecelerateInterpolator());

//挤压与拉伸，宽度变大

ValueAnimator squashAnim2 = ObjectAnimator.*ofFloat*(newBall,

"width", newBall.getWidth(),

newBall.getWidth() + 50);

squashAnim2.setDuration(duration/4);

squashAnim2.setRepeatCount(1);

squashAnim2.setRepeatMode(ValueAnimator.*REVERSE*);

squashAnim2.setInterpolator(**new** DecelerateInterpolator());

//挤压与拉伸，y变大

ValueAnimator stretchAnim1 = ObjectAnimator.*ofFloat*(newBall,

"y", endY,endY + 25f);

stretchAnim1.setDuration(duration/4);

stretchAnim1.setRepeatCount(1);

stretchAnim1.setInterpolator(**new** DecelerateInterpolator());

stretchAnim1.setRepeatMode(ValueAnimator.*REVERSE*);

//挤压与拉伸，高度变小

ValueAnimator stretchAnim2 = ObjectAnimator.*ofFloat*(newBall,

"height",

newBall.getHeight(),

newBall.getHeight() - 25);

stretchAnim2.setDuration(duration/4);

stretchAnim2.setRepeatCount(1);

stretchAnim2.setInterpolator(**new** DecelerateInterpolator());

stretchAnim2.setRepeatMode(ValueAnimator.*REVERSE*);

//上升

ValueAnimator bounceBackAnim = ObjectAnimator.*ofFloat*(newBall,

"y", endY, startY);

bounceBackAnim.setDuration(duration);

bounceBackAnim.setInterpolator(**new** DecelerateInterpolator());

// Sequence the down/squash&stretch/up animations

AnimatorSet bouncer = **new** AnimatorSet();

bouncer.play(bounceAnim).before(squashAnim1);

bouncer.play(squashAnim1).with(squashAnim2);

bouncer.play(squashAnim1).with(stretchAnim1);

bouncer.play(squashAnim1).with(stretchAnim2);

bouncer.play(bounceBackAnim).after(stretchAnim2);

// Fading animation - remove the ball when the animation is done

ValueAnimator fadeAnim = ObjectAnimator.*ofFloat*(newBall,

"alpha", 1f, 0f);

fadeAnim.setDuration(250);

fadeAnim.addListener(**new** AnimatorListenerAdapter() {

@Override

**public** **void** onAnimationEnd(Animator animation) {

balls.remove(((ObjectAnimator)animation).getTarget());

}

});

// Sequence the two animations to play one after the other

AnimatorSet animatorSet = **new** AnimatorSet();

animatorSet.play(bouncer).before(fadeAnim);

// Start the animation

animatorSet.start();