

The people



Motivation

- ▶ Create game 'levels' (procedurally produce game content) from the available music
- ▶ Explore (or rather experience) music collections visually
- ▶ Multiplayer game that fully uses the sensor range of mobile devices

Game engine

Innovative, platform independent

- ▶ Scene graphs
- ▶ Input
 - ▶ Multitouch
 - ▶ Motion control
- ▶ Manage music features extracted from songs
- ▶ Client/Server (RPC) support for multiplayer

Game

Minigame Engine

Game Engine

Engine -
Blackberry

Engine -
Android

BlackBerry

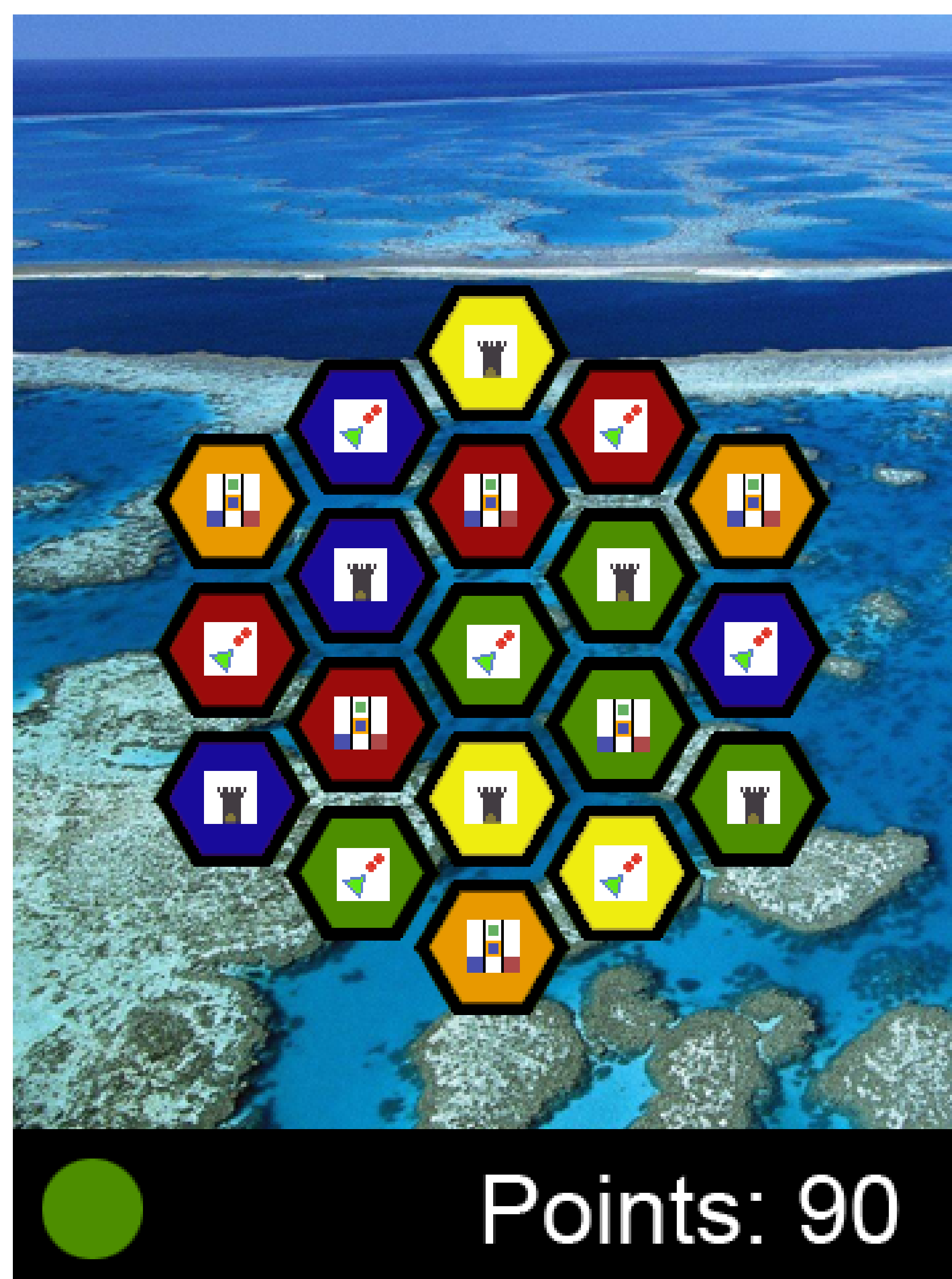


Game concept

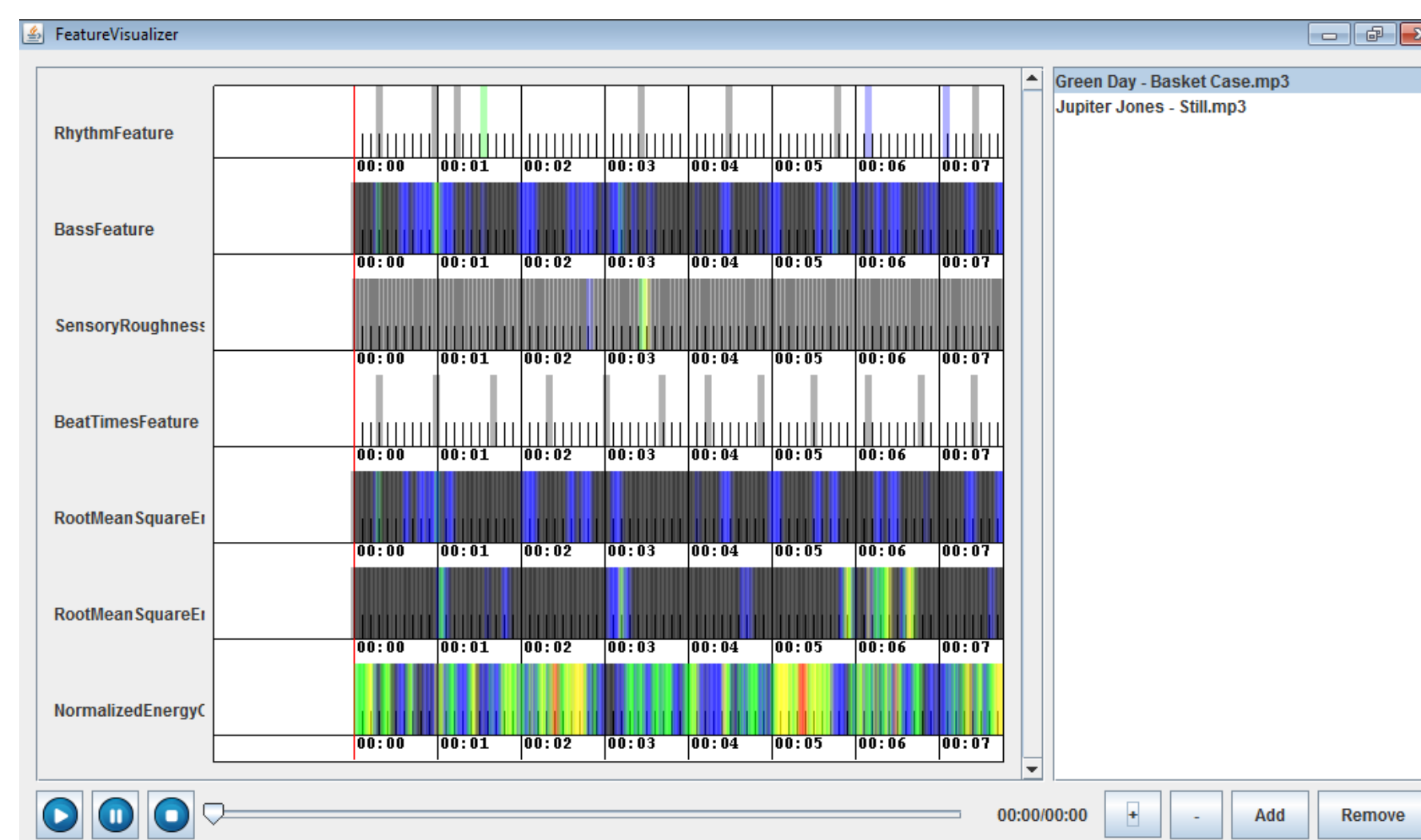
Global board and mini games

- ▶ Each field represents a song
- ▶ One minigame per field, fixed
- ▶ Conquer fields by achieving new high score
- ▶ Adjacent fields provide more points
- ▶ Completely asynchronous:
users play whenever, whatever they want

Board

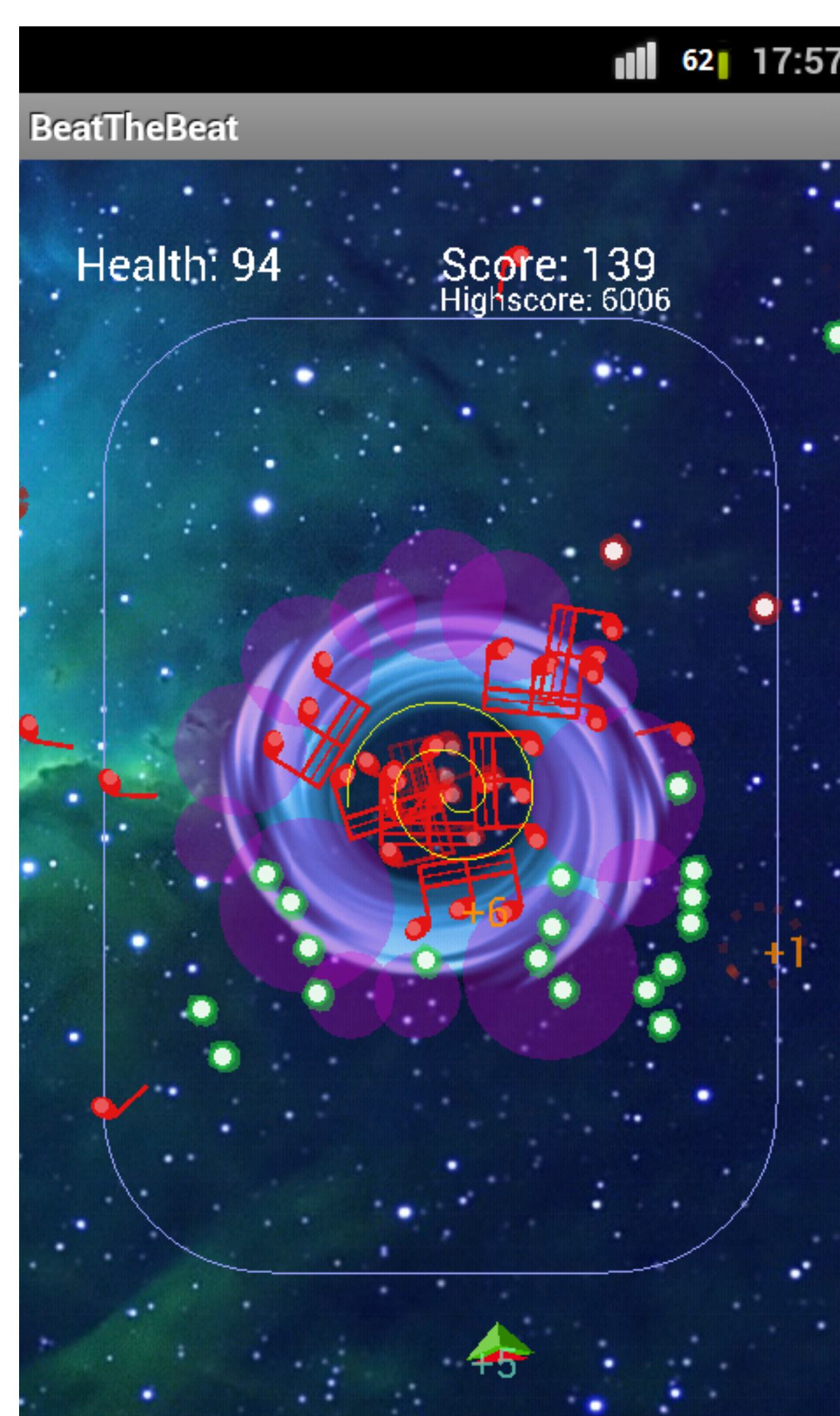


- ▶ Symbols correspond to mini games
- ▶ Each player has an unique color
- ▶ Board setup uses self-organizing maps on the extracted music features of the songs



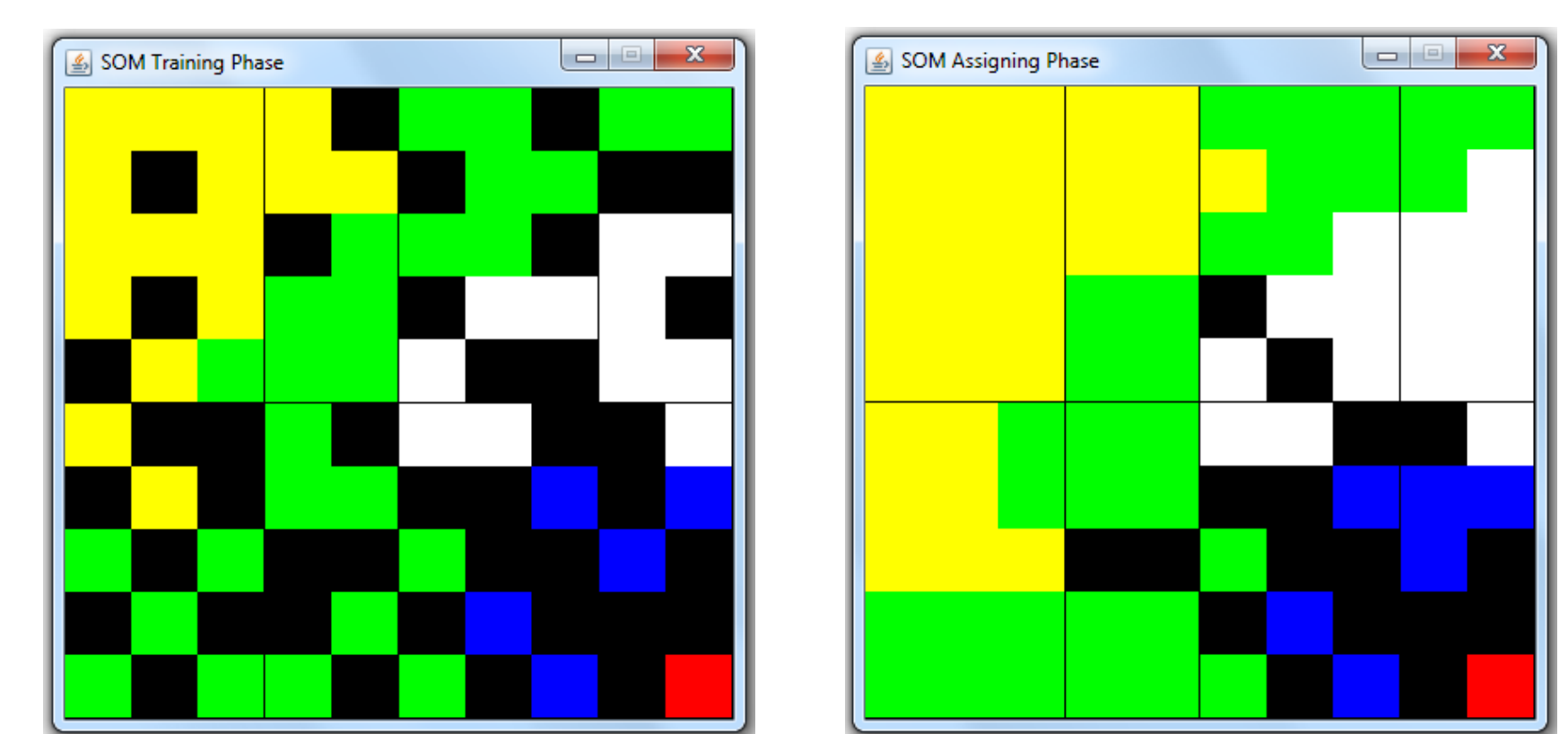
Feature visualizer for psycho-acoustic features

MusicFighter



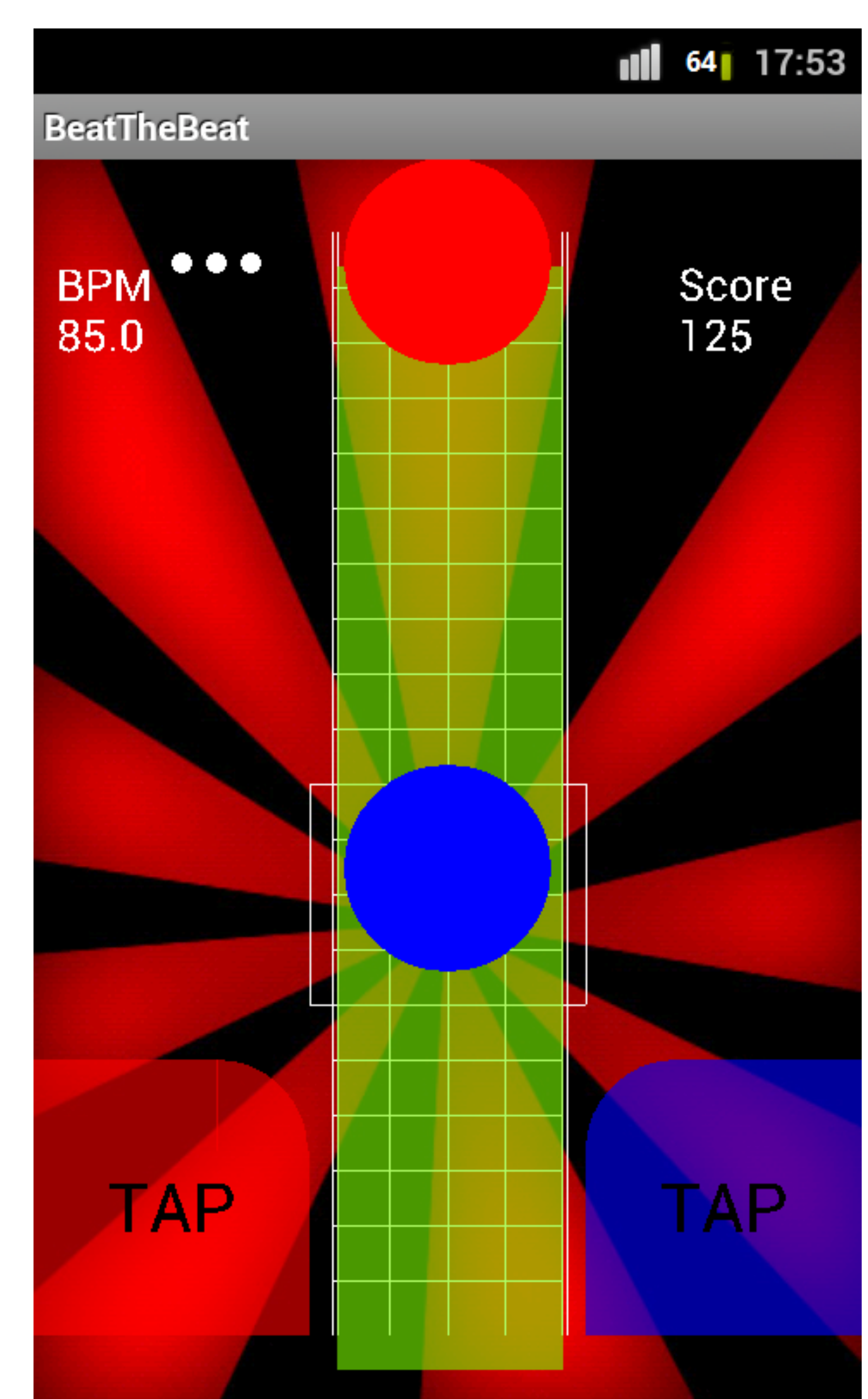
- ▶ Gameplay: shooter, 4 enemy types
- ▶ Music features control spawning times and enemy set composition

Self-organized song mapping



2 phases, left: initial feature-based distribution by SOM, right: after simple repair procedure

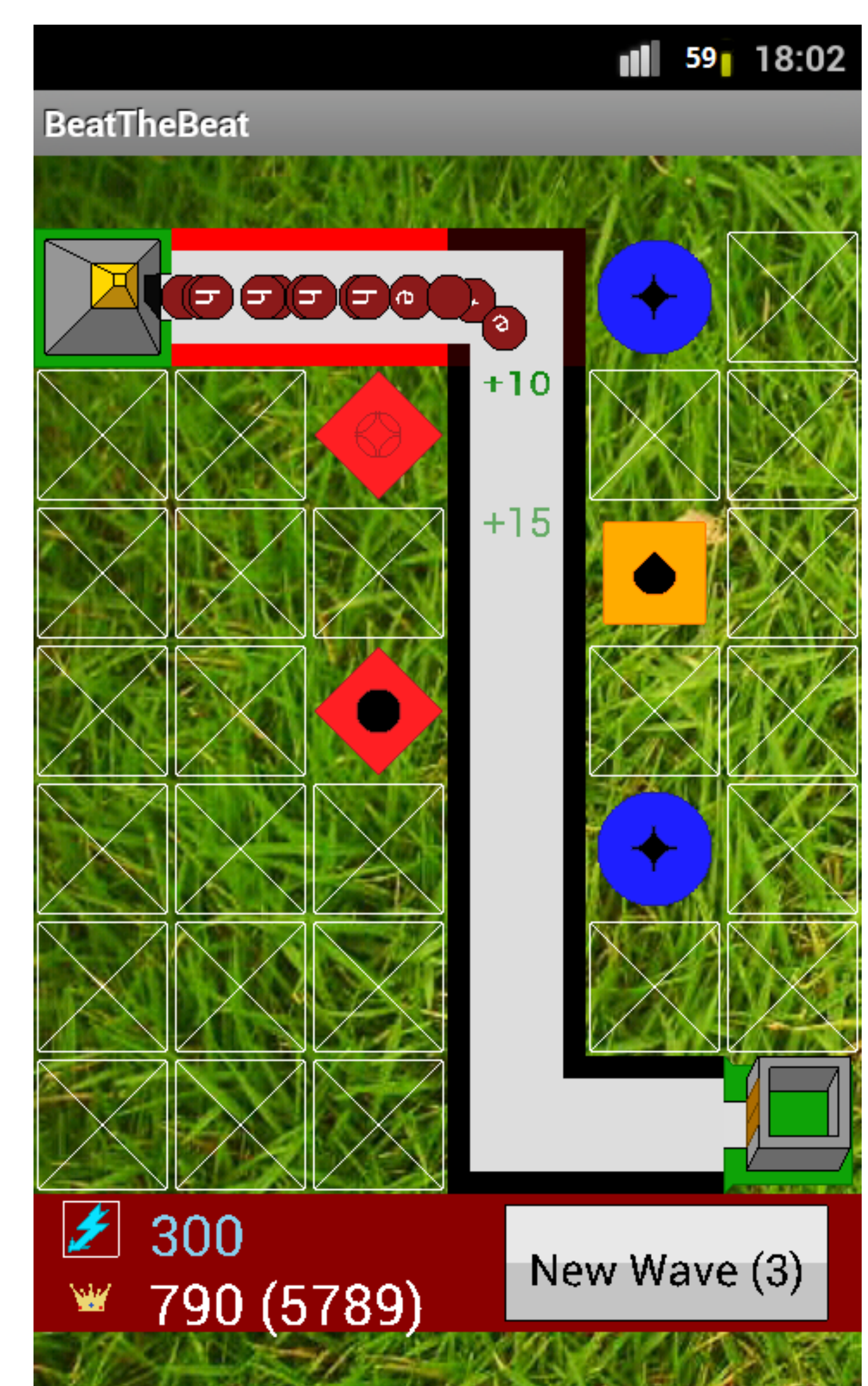
TapByTap



Rhythm 'tapping' game

- ▶ Color balls generated with response to rhythmic characteristics
- ▶ Getting more difficult during the song

TowerDefense



- ▶ Tower shot ranges and strengths controlled by music features
- ▶ It helps to know the music (training effect)