

用pygame写军旗游戏课题研究报告

- 研究背景
 - 简介
 - 背景
- 研究目的
- 问题探究
 - 需要的工具
 - 项目结构
 - 研究中的问题
- 总结
- 成果展示
 - 电脑端
 - 手机端
- 有待解决的问题
- 参考

用pygame写军旗游戏课题研究报告

研究背景

简介

我叫胡宗尧，这是我和我的弟弟合作的一个项目：用 *pygame*¹ 编写军旗游戏。

我对 *python* 有浅显的了解，并且用 *pygame* 做过一些小游戏。

我的弟弟叫胡宗禹，他从 *C* 语言入门，对图形界面不是特别了解，但接触过 *OpenGL*。

[源码下载地址](#)，提取码：n3g3

[字体下载地址](#)：提取码：4at1

没有安装*python* | *pygame*也想玩？

[EXE下载地址](#)，提取码：qano

背景

作为一个军旗爱好者，我经常在家和弟弟下军旗，翻翻棋²作为一种冷门的下法，在很多军旗的App中都没有得到实现，所以在这个课题中，我们就用 *pygame* 实现这个游戏。

研究目的

为了让更多人了解翻翻棋。能玩上军棋，爱上玩军棋。

问题探究

需要的工具

我们在这个项目中使用的工具如下：

- Sublime Test 3
- Windows 7
- python 3.5.1
- pygame 1.9.4

项目结构

```
1  # -*- coding: utf-8 -*-
2  # Time : 2019/1/29 21:39
3  # Author : hzy
4
5  #初始化pygame
6  import pygame,random
7  from pygame.locals import *
8  from sys import exit
9
10 pygame.init()
11 SCREEN_SIZE = (540,1080)
12 screen = pygame.display.set_mode(SCREEN_SIZE, 0, 32)
```

```
1  #所有的变量定义
2  font = pygame.font.Font("HanYiYanKaiW-2.ttf", 30)
3  font_height = font.get_linesize()
4  move=False
5  compare=False
6  red_turn=True
7  select_block=None
8  clock=pygame.time.Clock()
9
10 listfast=[(0,1),(1,1),(2,1),(3,1),(4,1),(0,2),(4,2),(0,3),(4,3),(0,4),
            (4,4),(0,5),(1,5),(2,5),(3,5),(4,5),(0,6),(1,6),(2,6),(3,6),(4,6),(0,7),
            (4,7),(0,8),(4,8),(0,9),(4,9),(0,10),(1,10),(2,10),(3,10),(4,10)]
```

```

11
12 list_qi=['军旗1', '司令1','军长1','师长1','师长1', '旅长1', '旅长1', '团长1',
13         '团长1',
14         '营长1', '营长1', '连长1', '连长1', '连长1', '工兵1', '工兵1', '工兵1', '排
15         长1',
16         '排长1', '排长1', '地雷1', '地雷1', '地雷1', '炸弹1', '炸弹1', '军旗0', '司
17         令0',
18         '军长0','师长0', '师长0', '旅长0', '旅长0', '团长0', '团长0', '营长0', '营长
19         0',
20         '连长0', '连长0', '连长0', '工兵0', '工兵0', '工兵0', '排长0', '排长0', '排
21         长0',
22         '地雷0', '地雷0', '地雷0', '炸弹0', '炸弹0']
23
24 list_circle=[(1,2),(3,2),(2,3),(1,4),(3,4),(1,7),(3,7),(2,8),(1,9),(3,9)]
25
26 list_block=[(0, 0), (0, 1), (0, 2), (0, 3), (0, 4), (0, 5), (0, 6), (0,
27 7), (0, 8), (0, 9), (0, 10),(0, 11), (1, 0), (1, 1), (1, 3), (1, 5), (1,
28 6), (1, 8), (1, 10), (1, 11), (2, 0), (2, 1), (2, 2),(2, 4), (2, 5), (2,
29 6), (2, 7), (2, 9), (2, 10), (2, 11), (3, 0), (3, 1), (3, 3), (3, 5), (3,
30 6),(3, 8), (3, 10), (3, 11), (4, 0), (4, 1), (4, 2), (4, 3), (4, 4), (4,
31 5), (4, 6), (4, 7), (4, 8),(4, 9), (4, 10), (4, 11)]
32
33 dAttackNumber={'司令':8, '军长':7, '师长':6, '旅长':5, '团长':4, '营长':3, '连
34 长':2, '排长':1, '工兵':0, '地雷':9, '炸弹':10, '军旗': 9}

```

```

1  #函数列表:
2  def toPygame(x,y,is_circle):
3
4  def fastGoWhere(pos,x,y,color,is_gb,times):
5
6  def whereToGo(pos):
7
8  def drawBlock():
9
10 def line(pos1,pos2):
11
12 def drawAll():
13
14 def select():
15
16 def change():
17
18 def moveqi():
19
20 def isbigger(chiqiText,beichiqiText):
21
22 def eatqi(text):

```

```

1  #初始化dictionary
2  # -----init-----
3  d={}
4  for x in range(-1,5):
5      for y in range(12):
6          a=(x,y)
7          if a in list_block:
8              i=random.randint(0,len(list_qi)-1)
9              d[a]=[list_qi[i][-1],list_qi.pop(i)[0:2],0]
10         else:
11             d[a]=[ ]
12
13 pygame.display.set_caption("红方走棋")
14 drawAll()
15 drawBlock()
16 # -----end===init-----
17
18 #主循环
19 while True:
20     for event in pygame.event.get():
21         if event.type == QUIT:
22             exit()
23         if event.type == MOUSEBUTTONDOWN:
24             pos = pygame.mouse.get_pos()
25             a=(pos[0]//120,pos[1]//90)
26             qi=d[a]
27             drawAll()
28             if qi==[ ]:
29                 if select_block!=None:
30                     drawBlock()
31                     whereToGo(select_block)
32                     if screen.get_at(pos)==(0,255,0):
33                         select()
34                         moveqi()
35                     elif screen.get_at(pos)==(255,0,0):
36                         select()
37                         eatqi(qi[1])
38                 else:
39                     select_block=None
40                     drawAll()
41             elif qi[2]:
42                 if select_block==None:#无选中棋
43                     if ((red_turn and qi[0]=="0") or ((not red_turn) and
qi[0]=="1")) \
44                         and qi[1]!="地雷" and qi[1]!="军旗":
45                         select_block=a
46                         select()
47                 else:#有选中棋

```

```

48         drawBlock()
49         whereToGo(select_block)
50         if ((qi[0]=="0" and red_turn) or (qi[0]!="0" and not
red_turn)) \
51             and qi[1]!="地雷" and qi[1]!="军旗":
52             select_block=a#切换选中的棋
53             select()
54             drawAll()
55             select()
56             elif screen.get_at(pos)==(255,0,0):
57                 select()
58                 eatqi(qi[1])#吃棋
59             elif screen.get_at(pos)==(0,255,0):
60                 select()
61                 moveqi()
62             else:
63                 select_block=None
64                 drawAll()
65         elif select_block==None:
66             qi[2]=1
67             select_block=None
68             change()
69         else:
70             whereToGo(select_block)
71             if screen.get_at(pos)==(255,0,0):
72                 qi[2]=1
73                 if qi[0]==d[select_block][0]:
74                     change()
75                     select_block=None
76                     drawAll()
77                 else:
78                     select()
79                     eatqi(qi[1])#吃棋
80             else:
81                 select_block=None
82                 drawAll()
83         drawBlock()
84         whereToGo(select_block)
85         if red_turn:
86             screen.blit(font.render("红方", True, (255,0,0)),
(240,1050))
87         else:
88             screen.blit(font.render("黄方", True, (255,255,0)),
(240,1050))
89
90         if move:
91             drawAll()
92             drawBlock()
93             pos1=(pos1[0]+step_x,pos1[1]+step_y)

```

```

94         pygame.draw.rect(screen, (255,255,255), Rect(pos1,(60,60)),0)
95         screen.blit(font.render(text, True, (255,255*color,0)), pos1)
96         if abs(float(pos1[0]-pos2[0]))<=abs(step_x) and
abs(float(pos1[1]-pos2[1]))<=abs(step_y):
97             move=False
98             if compare:
99                 if isbigger(text,beichiqiText)>0:
100                     d[a]=[str(color),text,1]
101                 elif isbigger(text,beichiqiText)==0:
102                     d[a]=[ ]
103                 elif isbigger(text,beichiqiText)<0:
104                     pass
105                 compare=False
106             else:
107                 d[a]=[str(color),text,1]
108             drawAll()
109             drawBlock()
110         pygame.display.update()
111         # clock.tick(20)

```

研究中的问题

1.安装pygame时屡次出错

解决方案：在网上查阅很多资料得知要用 1.9.4 版本的 pygame，问题解决

2.工兵移动的问题

众所周知，工兵的移动非同寻常棋子-----它在轨道上可以拐弯，这就增加了很多难度。

解决方案：我使用了递归语句判断是否达到拐点，问题解决。代码如下：

```

1  def fastGoWhere(pos,x,y,color,is_gb):
2      try:
3          if is_gb:
4              for a,b in [(-1,0),(0,1),(0,-1),(1,0)]:
5                  if (a,b)!=(-x,-y):
6                      nextPos=(pos[0]+a,pos[1]+b)
7                      nextQi=d[nextPos]
8                      if nextPos in listfast:
9                          if nextQi==[]:
10                             pygame.draw.circle(screen, (0,255,0),

```

```

11         toPygame(*nextPos,True), 15, 0)
12         fastGoWhere(nextPos,a,b,color,True)
13         elif (not nextQi[2]) or (nextQi[0]!=color):
14             pygame.draw.circle(screen, (255,0,0),
15                 toPygame(*nextPos,True), 15, 0)
16     else:
17         nextPos=(pos[0]+x,pos[1]+y)
18         nextQi=d[nextPos]
19         if nextPos in listfast:
20             if nextQi==[]:
21                 pygame.draw.circle(screen, (0,255,0),
22                     toPygame(*nextPos,True), 15, 0)
23                 fastGoWhere(nextPos,x,y,color,False,times+1)
24             elif (not nextQi[2]) or (nextQi[0]!=color):
25                 pygame.draw.circle(screen, (255,0,0),
26                     toPygame(*nextPos,True), 15, 0)
27     except KeyError:
28         pass
29
30 def whereToGo(pos):
31     if pos!=None:
32         pygame_pos=toPygame(*pos,True)
33         for x in [-1,0,1]:
34             for y in [-1,0,1]:
35                 try:
36                     if
screen.get_at((pygame_pos[0]+x*44,pygame_pos[1]+y*33))==(0,0,255):
37                     nearbyPos=(pos[0]+x,pos[1]+y)
38                     nearbyQi=d[nearbyPos]
39                     qi=d[pos]
40                     if nearbyQi==[]:
41                         if nearbyPos in listfast and pos in listfast:
42                             fastGoWhere(nearbyPos,x,y,qi[0],qi[1]=='I
兵',1)
43
44                     pygame.draw.circle(screen, (0,255,0),
45                         (pygame_pos[0]+120*x,pygame_pos[1]+90*y),
46                         15, 0)
47                     elif (not nearbyQi[2] or nearbyQi[0]!=qi[0]) and
(nearbyPos not in list_circle):
48                         pygame.draw.circle(screen, (255,0,0),
49                             (pygame_pos[0]+120*x,pygame_pos[1]+90*y),
50                             15, 0)
51                     except IndexError:
52                         pass

```

3.递归可能造成的堆栈错误

如果工兵走的路线正好是一个圈时，递归就会永无止境-----直至报错。

解决方案：多加了一个times变量用来计数，每次传递到下一个函数时就加一，大于40时就退出递归。

代码很简单：

```
1 def fastGoWhere(pos,x,y,color,is_gb,times):
2     try:
3         if is_gb and times<40:
4             for a,b in [(-1,0),(0,1),(0,-1),(1,0)]:
5                 if (a,b)!=(-x,-y):
6                     nextPos=(pos[0]+a,pos[1]+b)
7                     nextQi=d[nextPos]
8                     if nextPos in listfast:
9                         if nextQi==[]:
10                             pygame.draw.circle(screen, (0,255,0),
11                                 toPygame(*nextPos,True), 15, 0)
12                             fastGoWhere(nextPos,a,b,color,True,times+1)
13                         elif (not nextQi[2]) or (nextQi[0]!=color):
14                             pygame.draw.circle(screen, (255,0,0),
15                                 toPygame(*nextPos,True), 15, 0)
16                     elif times<40:
17                         nextPos=(pos[0]+x,pos[1]+y)
18                         nextQi=d[nextPos]
19                         if nextPos in listfast:
20                             if nextQi==[]:
21                                 pygame.draw.circle(screen, (0,255,0),
22                                     toPygame(*nextPos,True), 15, 0)
23                                 fastGoWhere(nextPos,x,y,color,False,times+1)
24                             elif (not nextQi[2]) or (nextQi[0]!=color):
25                                 pygame.draw.circle(screen, (255,0,0),
26                                     toPygame(*nextPos,True), 15, 0)
27             except KeyError:
28                 pass
29
30 def whereToGo(pos):
31     if pos!=None:
32         pygame_pos=toPygame(*pos,True)
33         for x in [-1,0,1]:
34             for y in [-1,0,1]:
35                 try:
36                     if
37 screen.get_at((pygame_pos[0]+x*44,pygame_pos[1]+y*33))== (0,0,255):
38                     nearbyPos=(pos[0]+x,pos[1]+y)
39                     nearbyQi=d[nearbyPos]
40                     qi=d[pos]
41                     if nearbyQi==[]:
```



```
41         if nearbyPos in listfast and pos in listfast:
42             fastGoWhere(nearbyPos,x,y,qi[0],qi[1]=='工
兵',1)
43             pygame.draw.circle(screen, (0,255,0),
44                 (pygame_pos[0]+120*x,pygame_pos[1]+90*y),
15, 0)
45             elif (not nearbyQi[2] or nearbyQi[0]!=qi[0]) and
(nearbyPos not in list_circle):
46                 pygame.draw.circle(screen, (255,0,0),
47                     (pygame_pos[0]+120*x,pygame_pos[1]+90*y),
15, 0)
48             except IndexError:
49                 pass
```

4.打包成EXE中出现的问题

报错：应用程序异常退出

解决方法：

查阅了资料得知，是字体没有找到的原因。

只要把我的字体模块 `HanYiYanKaiW-2.ttf` 放到源码所在的目录下即可

总结

这个军棋游戏还是费了我很多心思的。但是做出来后感到很开心。

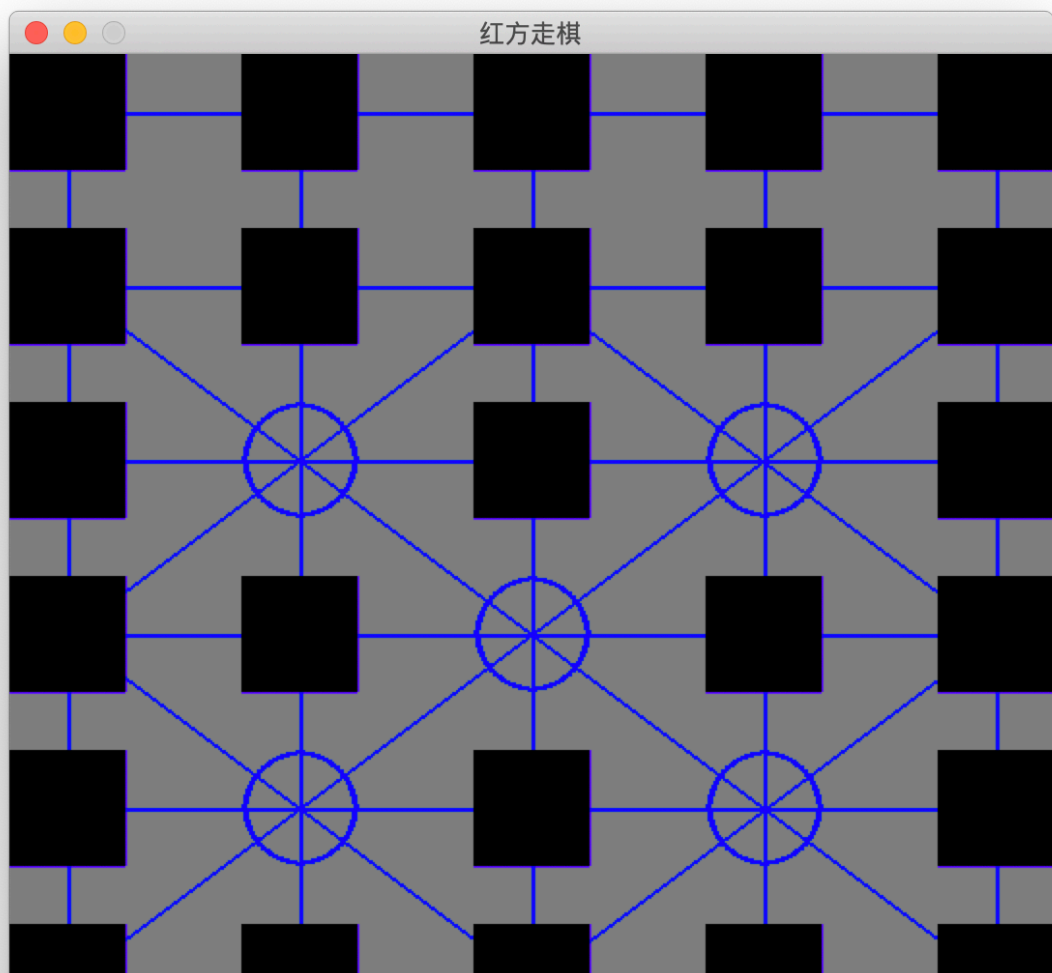
成果展示

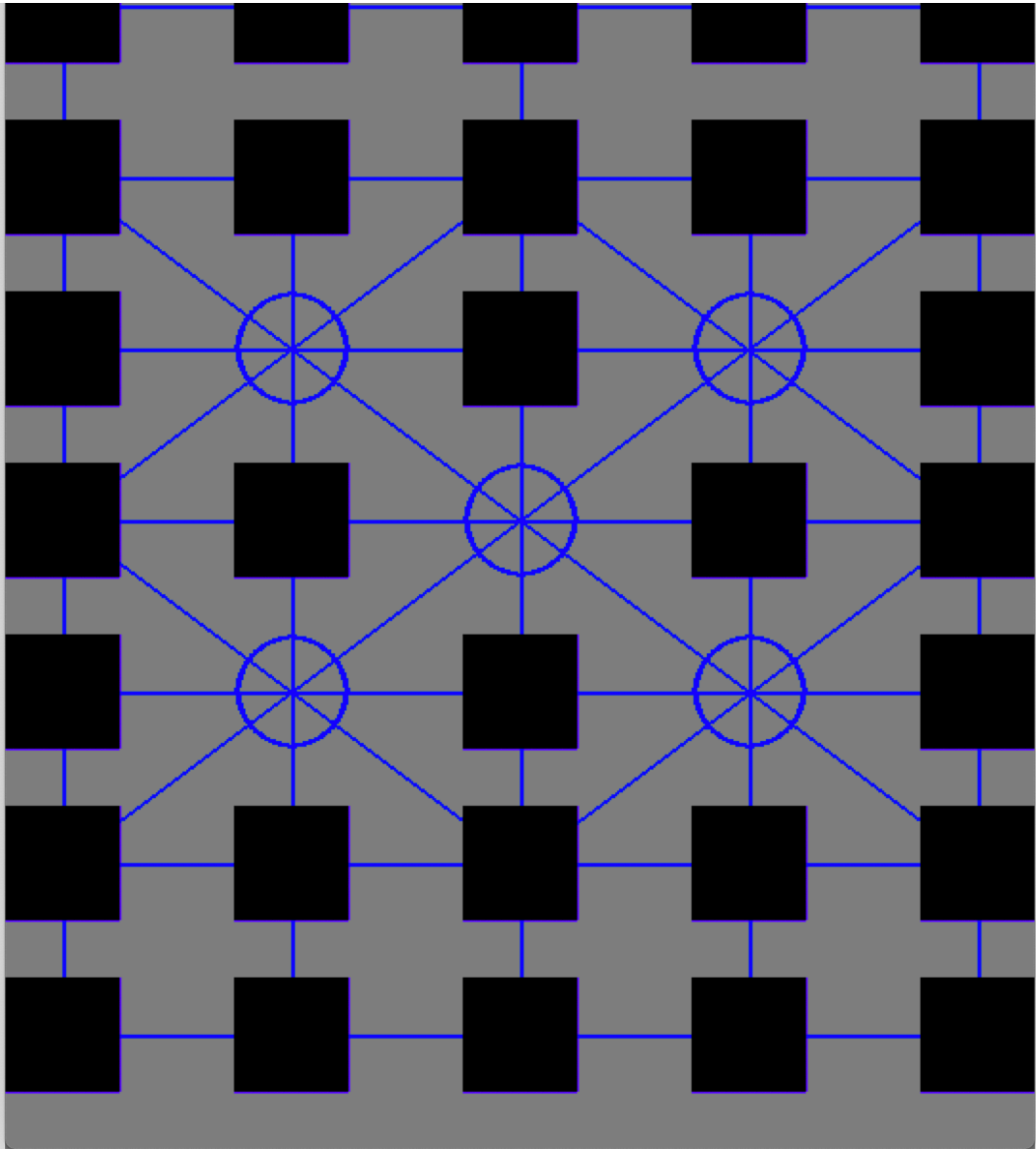
电脑端

这是用 `pyinstaller` 打包后的游戏界面有个大大的【军棋】框：

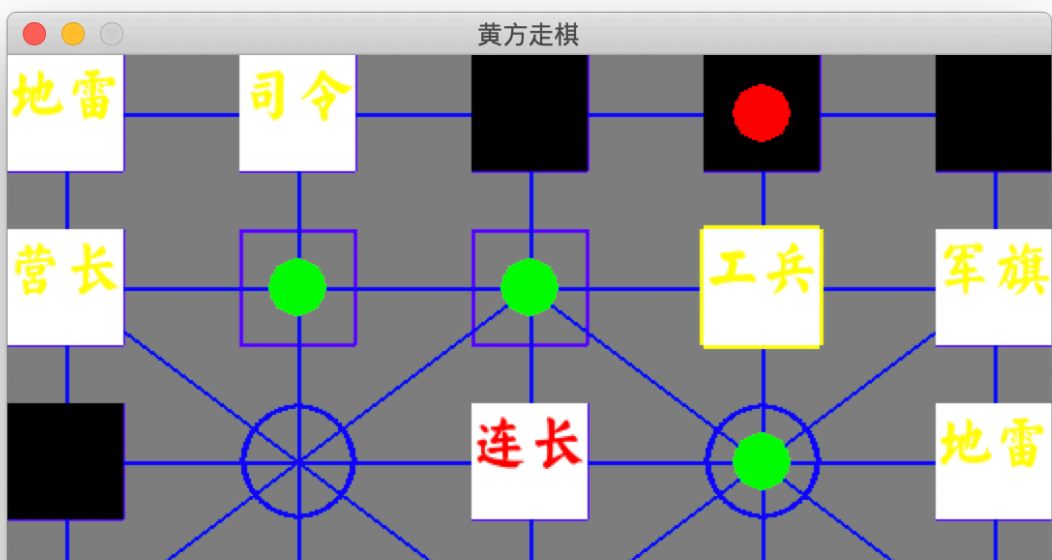


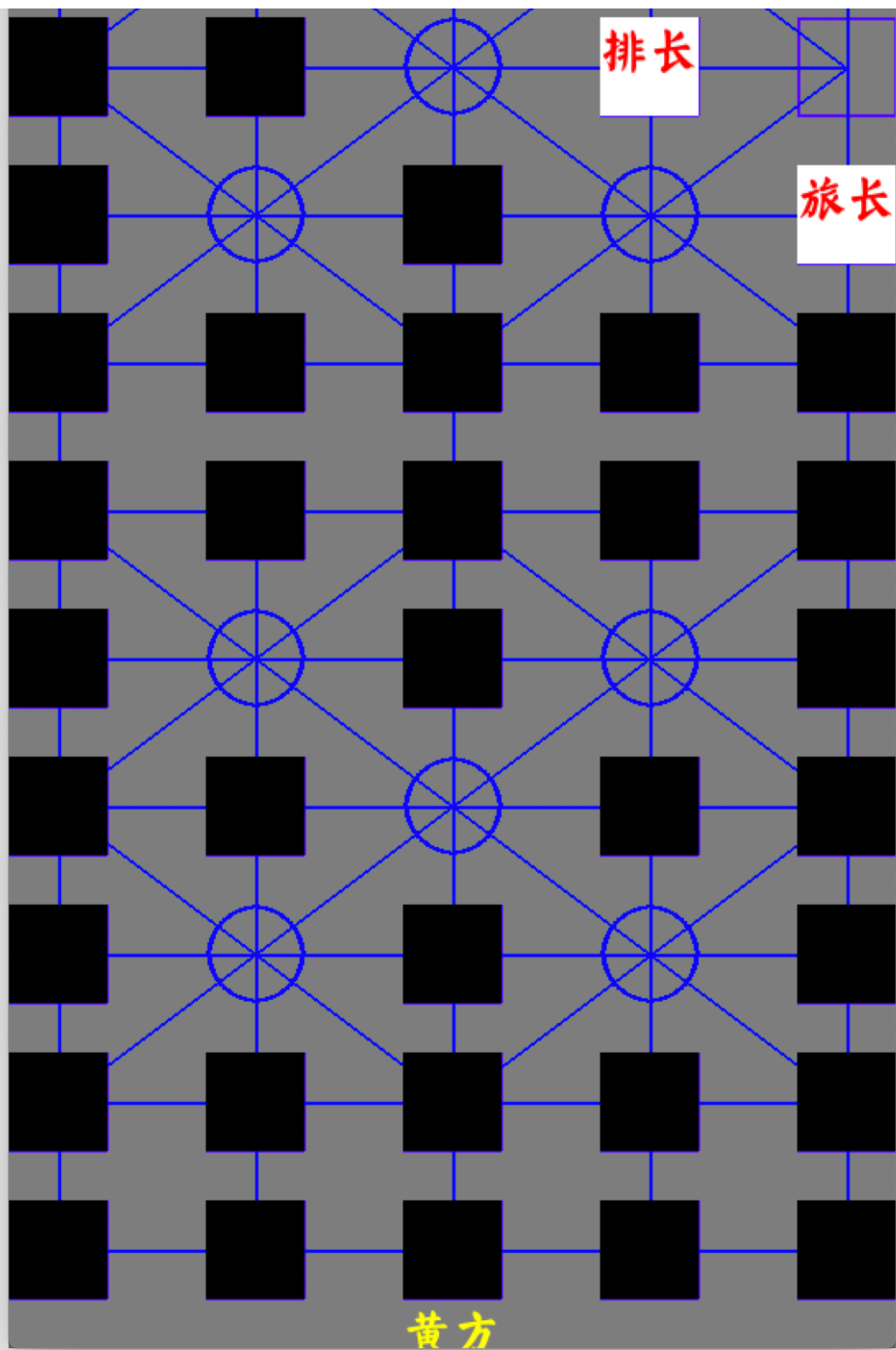
点了【军棋】后会出现军棋界面：





然后就可以愉快的玩耍啦！！！！！！！！！！





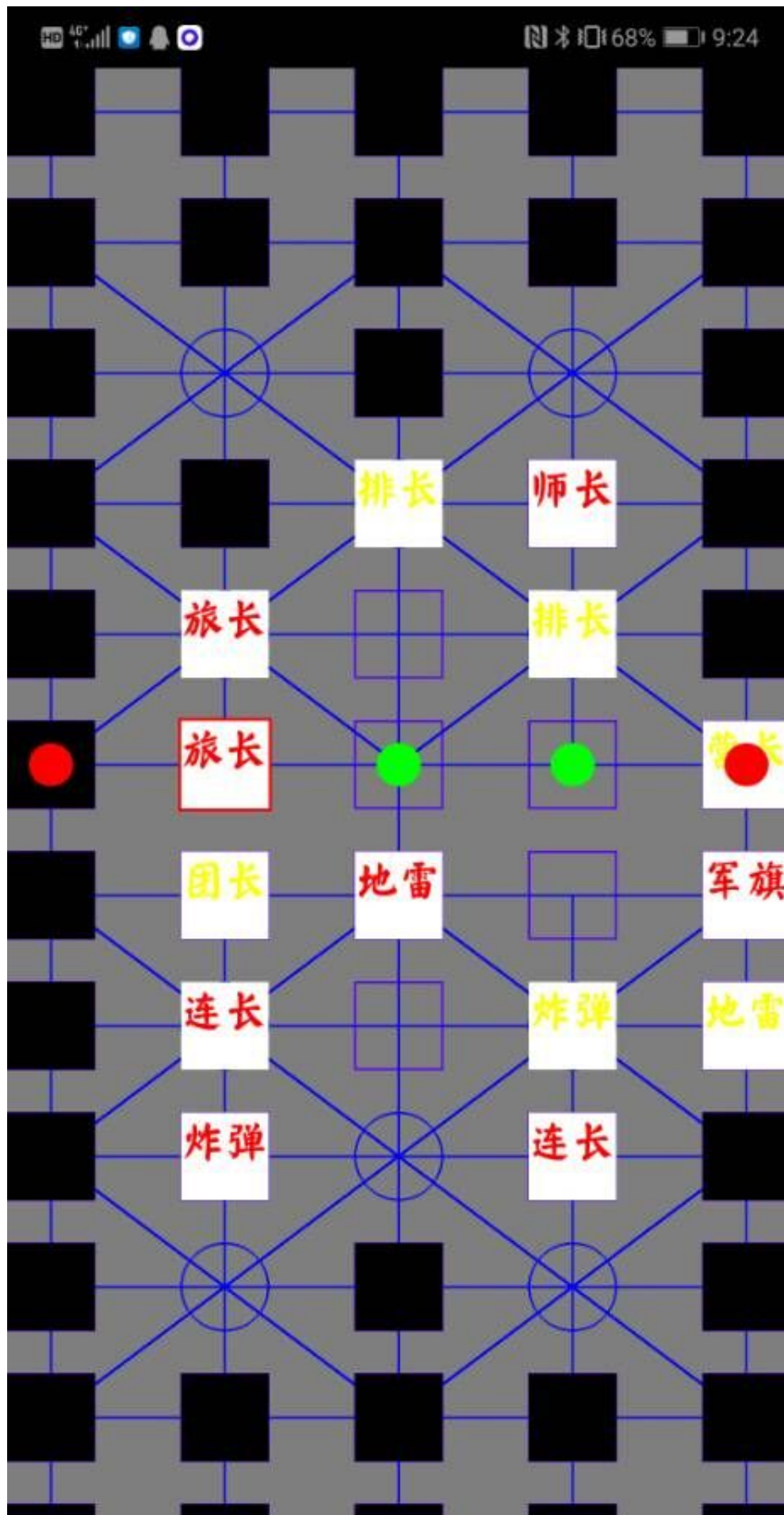
手机端

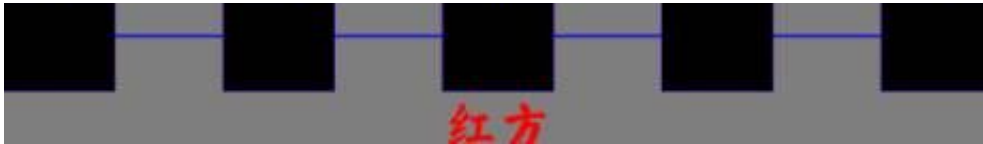
在手机端玩翻翻棋的方法：

1. 下载pydroid
2. 在pydroid上安装pygame模块

3. 导入 *junqi.py* 文件

4. 运行





有待解决的问题

- 被吃掉的军棋无法查看的问题-----我会马上解决的!
- 最好能弄一个人机对战的军棋游戏-----这就有点难度了，我最近正好在研究深度学习。相信不久就能训练出来能够进行人机对战的
- EXE文件只能在win7上运行-----暂时没有时间，但我也想到了一个办法。运行一个win10虚拟机，打包成适合win10的EXE文件。然后在MacBook Pro上也打包一遍，就有了一个Unix可执行文件了。

参考

[目光博客：一个IT人的清静小后院](#)

[维基百科](#)

1. 是跨平台Python模块，专为电子游戏设计。包含图像、声音。创建在SDL基础上，允许实时电子游戏研发而无需被低端语言，如C语言或是更低端的汇编语言束缚。基于这样一个设想，所有需要的游戏功能和理念都（主要是图像方面）完全简化位游戏逻辑本身，所有的资源结构都可以由高级语言提供，如Python。-----来自维基百科 [↗](#)

2. 规则说明：2人参与游戏，玩家坐下后，会根据玩家角色性别随机选定军棋翻翻棋的角色秀。若当桌的两个玩家为同性，则角色分别为不同的2个形象。开始游戏后，2人轮流将棋盘上反面放置的军棋翻起以确定控制方，直到某一玩家先翻到与前一张翻到棋子颜色相同，则该棋子的颜色就属于该玩家；开棋方由系统随机选取。棋子分为两色，一共50颗棋子，每方各有25个棋子，分别是：3个工兵，3个排长，3个连长，2个营长，2个团长，2个旅长，2个师长，1个军长，1个司令，2个炸弹，3个地雷，1个军旗 大小：司令 > 军长 > 师长 > 旅长 > 团长 > 营长 > 连长 > 排长 > 工兵地雷：地雷不能移动。工兵能挖或者炸弹能炸，其他棋子都不能主动触碰地雷。军旗：不能移动。当对方3个地雷全被挖后，可以用本方任何棋子去扛军旗，挖掉军旗则获胜。炸弹：可以移动。能炸掉对方任何棋子（同归于尽），若对方地雷未被挖光，则无法碰撞对方军旗。其他可以移动棋子可以主动触碰炸弹以同时消失。-----来自百度百科 [↗](#)