SMART CAMERA

ver. 1.0

by Rydygier

1. DESCRIPTION

Smart Camera module provides an autonomous, dynamic camera floating over whole map that, hopefully in smart way, will on its own decide, what is most interesting to show at the moment, and how to show it. SC module applies also some subtle post processing effects, simulation of hand shaking, manual zoom and focus adjustment – all switchable. SC can be launched at mission init or at chosen moment, can be also terminated any time. Apart of that SC, when running, offers quick switch from camera view to normal view and back. SC is not interactive – camera can't be controlled manually, is fully controlled by the algorithm. To work needs some units on map to watch. It is always focused on some unit or vehicle.

Smart Camera can be used for dynamic (unique each run) intros, outros, cut scenes and such. May be useful also for showcases, FPS measurement, even as alternative death cam mode.

Note, SC relies on SQF code. That means, SC will be affected by consequences of overloaded scheduler/CPU. Be sure, that yours CPU can seamlessly handle all used scripts, addons and such before use.

SC comes in two versions: as "Smart Camera" module, that can be placed on map in editor, and as a script, so more advanced users can alter the algorithm itself as needed.

2. INSTALLATION

Module version: put ***@RYD_Scam*** folder same place, where all other addons are stored, and play with addon. Now new module should be available in editor.

Script version: put SmartCam folder in your mission folder. To init use such code in the init.sqf file or init field of any object on map:

```
nul = [] execVM "SmartCam\CamInit.sqf";
```

3. CONFIGURATION

Both versions allow to configure certain features.

Initial position

By default camera will start looking north 45 meters over first observed unit (script version) or over position of SC module (module version).

You can point custom object (eg empty trigger) as initial position by naming it **RydSC_InitPos**.

For module version only there is alternative possibility. You can set on map any number of SC modules you like. SC will choose randomly one of them as init position.

To change default init level, put this code into init field of object chosen as initial position:

this setVariable ["RydSC_InitLevel",level];

(where “level” is chosen number in meters).

To change default init looking direction same way use such code:

this setVariable ["RydSC_InitDir",direction];

(where “direction” is chosen value in degrees from 0 to 360).

Black list

You can point chosen groups as blacklisted, so never chosen by camera. Do it by defining at init such global variable:

RydSC_BLGroups = [(names of excluded groups here)];

Example: RydSC_BLGroups = [(group player),Group1];

This variable can be updated anytime.

Priority groups

You can choose some groups to be more frequently chosen, than other:

RydSC_Prior = [value, (list of group names)];

(where “value” means multiplier of “bait” value for each of listed groups. 1 will change nothing, 2 will double chances of listed groups to be chosen, 0.5 will half it).

Example: RydSC_Prior = [2,Group1,Group2,Group3];

This variable can be updated anytime.

Custom shots pool

SC by default randomly chooses next shot (position relative to watched target) from the list of shots parameters in form of relative positions arrays. If you want, you can define variable, that will contain yours own pool of positions, same as default is:

```
RydSC_RelPos =
        [
        [0,-2,1.5],
        [0,-3,1.75],
        [0,-5,2.0],
        [0,-60,25],
        [60,0,25],
        [0,60,25],
        [-60,0,25],
        [3,0,1.75],
        [5,0,2.0],
        [-3,0,1.75],
        [-5,0,2.0],
        [-3,-3,1.75],
        [3,-3,1.75],
        [-3,3,1.75],
        [3,3,1.75],
        [5,-5,2.0],
        [-5,5,2.0],
        [5,5,2.0]
        ];
```

(note, that final position can be different, for vehicles values are multiplied to keep proper distance, also sometimes SC may calculate autonomously own shot position).

Additional effects

SC has active by default subtle post processing effects and other additional effects, like "shaking hand", zooming and manual focusing adjustments simulation. To switch them off/on use:

```
RydSC_PPEffects = false/true;
```

```
RydSC_Additional = false/true;
```

These variables can be updated anytime.

Delaying SC launch

To do that, define before launching (at init) such variable:

```
RydSC_WaitForMe = false;
```

After that SC will wait with launching until above variable will be set as true (*RydSC_WaitForMe = true;*).

Disabling view switch

If you do not want any view switching, use:

RydSC_ViewSwitch = false;

This variable can be updated anytime.

Setting minimal transitions height

By default height of camera movement between targets is equal to maximal ground height between these targets plus 3 meters. That additional value can be changed anytime by using:

RydSC_TransHeight = value;

(where "value" is desired height in meters. Should be positive).

Tweaking chance for longer focus on the chosen target

For this purpose use anytime:

RydSC_ChangeTendMpl = value;

(where "value" is a number, by default equal to 1 multiplier of chance for target change next cycle (checked every 5-10 seconds). The higher value, the higher chance of changing target every cycle).

Terminating SC

To completely end working of the code (including removing from map all SC modules, if used), after its launch define such variable:

RydSC_Active = false;

After that, to make SC working again, whole code must be reinitialized.

4. USAGE

Just watch and enjoy. ☺

Module version folder contains also very simple demo mission for it. That may be good start. See what way things are set up there. This is only one exemplary way of course. In that demo to start SC use *Juliet* radio channel (0-0-0). Second trigger after 60 seconds from mission start will automatically disable SC.

When SC view active, you can press any key on keyboard for quick switch to the normal view. Unless terminated, camera is still flying out there, so any moment you can switch back by using action menu option.

To see any post processing effects, should be enabled this effects presence in game's video advanced settings. For focusing effects minimal setting is "Normal".

Suggestion: in some cases may be good idea to combine SC view mode with music. To do so, just put on map JukeBox module, if you want the music from the beginning, or spawn it at chosen moment:

```
_gpL = creategroup sideLogic;
_jb = _gpL createUnit ["JukeboxManager", [1000,1000], [], 0, "NONE"];
```

(use variables without initial underscores (_) if code should be used from init/act field, not from SQF file).

To set up alternative death cam mode using SC, just put whole content of ***Script version*** folder (including ***onPlayerKilled.sqs*** file) into your mission folder. Then simply die playing that mission to test it. Using simultaneously with activated SC module is possible, but not recommended due to extended delay (SC must exit before death camera mode launch, which can take some seconds and cause small visual issues at init).

5. HOW IT WORKS

The point is, how camera selects objects to observation. For each group leader on map (or his vehicle) camera each cycle calculates in sophisticated way so called "bait factor". The bigger number, the bigger chance for choosing that object. "Bait factor" increases with object speed, number of nearby units, count of fired nearby shots and count of hits. Decreases with time, also when this object was chosen recently (to avoid stick the camera all the time with only one, very attractive target). When target is chosen, camera chooses also shot at random, but trying to keep not obstructed line of sight. If target knows about any enemy, camera will look for closest, and, if LOS between target and his enemy is keeped, will try to set such relative position (shot), so both will be in the screen frame to boost chance for most interesting point of view during firefight. When all this work is done, smooth camera transition from old position to new is performed. And so on.

Shaking effect is achieved by using invisible object, that in fact is the only true target of camera, and is moved from target to target. That object is "shaked" by special code.

There are known some issues.

Sometimes in effect LOS may be temporary blocked, if chosen target moved such way, so there is an obstacle between camera and "spied" object. Rather not avoidable in reasonable way.

Sometimes camera movements aren't that smooth, as I would like. Some reasons unknown, in other cases this is because observing very fast objects, like jets, so code not always can fully keep up, especially when CPU is overloaded.

Thanks for using Smart Camera

Rydygier