Third Person Cover Shooter 1.4

Video tutorials

Character setup : https://youtu.be/8aCr4eIMjFc
Add Pistol : https://youtu.be/CIes-mAj8q8
Add Rifle: https://youtu.be/HwN4rB-jPvo
AI setup: https://youtu.be/aliPWo91CZ4

Covers

Walls usable for taking covers have to be marked by cover markers. A cover marker is any game object with Cover and Box Collider components attached. Markers can intersect, form a chain and act as if that chain is one big cover.

Cover orientation matters, the example scene contains markers with feet that mark facing directions.

There are two kinds of covers, low and tall. The kind is determined from the height of a BoxCollider attached to the marker. The height threshold is different for every character and is defined by a Character Motor. In a chain of covers of different height character will correctly change its stance when transitioning between tall and low covers.

A corner of a cover with no adjacent covers nearby is treated as a corner characters can peek from. However, there can be unmarked walls and therefore character can attempt to take a peek in impossible situations. Such cases are handled by Open Left and Open Right

properties inside the Cover component, setting a value to false marks that corner as unusable for peeking.

Low covers can be climbed or vaulted over. Every cover has the type of climbing defined in a Cover component. Vaulting should be enabled for low walls players can jump over and climbing for covers alongside higher ground.

Character Motor

Characters must have a Character Motor component attached. It manages the character, it's movement, appearance and use of weapons. It handles gravity and therefore gravity should be turned off in the RigidBody component to avoid conflicts.

There is an IK (inverse-kinematics) system that handles aiming and recoil. It can be configured manually but for ease of use there is a button to set it up automatically inside the CharacterMotor inspector. It is recommended to reduce the amount of bones used by IK on non-player characters for performance reasons.

IK is calculated by adjusting bones until certain objects reach defined targets. Target objects must be part of the skeleton in order for changes to modify their transform. Character's sight is usually defined by a marker object that is part of the head, bones are transformed until marker's forward vector points towards the target.

Each character has a set of weapons in its disposal. To add a new weapon to the character you must create an object with a 3D model and a Gun component and attach it to a hand. Additionally, you can create a version of the weapon that is put into its holster. The motor automatically enables and disables weapon and holster objects.

Characters can throw grenades from both hands. Left hand is used in some situations when the character is hiding behind a cover. Grenades are cloned when thrown.

A character can be setup to have many hitboxes for various body parts. Setup is done inside Character Health component.

Gun

Guns raycasts bullet, manage clip and recoil.

For player characters bullets originate at camera in order for player to be able to fire on targets they can see, even if there is a small obstacle in front of the gun. The fire origin is set by a camera. Since AI do not have Third Person Camera attached their bullets are fired starting from the Aim marker, which usually is at the end of the gun.

Each weapon has two marker objects. Aim defines point of origin for AI bullets and is also used when rotating character's arms till the marker points towards the target. Left Hand object marks the position for character's left hand. Naming of left and right hands is incidental and character's handedness can be swapped.

The intended position of the left hand might differ in some animations, to handle that there are left hand marker overwrites you can use to set up IK for the left hand for some specific situations. Empty values are not used as overwrites.

Currently there are two kinds of weapons, pistols and rifles. The type defines character animations when using a weapon.

Grenade

Manage flight and explosion of a grenade. Flight and collisions are performed using raycasts. Grenades can bounce. Upon explosion a prefab is instantiated on grenades location. Grenade affects Character Health and Body Part Health objects in its explosion area. Damage is highest in the epicentre and decreases linearly to zero on the edge of the explosion area.

Third Person Controller

Takes player keyboard and mouse input and translates that to Character Motor commands.

Weapons are controlled by the number keys. Pressing 1 hides a weapon and keys starting with 2 make the character equip a weapon. The order of keys and weapons defined in the Character Motor is the same.

Enters grenade mode when pressing G. Displays a grenade preview using prefabs containing Path Preview and Explosion Preview components.

Mobile Controller

Takes touch screen input and translates that to Character Motor commands. Player input is generated using canvas objects containing components like Touch Movement or Touch Aiming. Those canvas components must be linked with the Mobile Controller.

When throwing a grenade, displays a grenade preview using prefabs containing Path Preview and Explosion Preview components.

Actor

Each character inside the level must have this component as the AI only regards objects with Actor as characters.

- Side: Team the actor is on. By default it's zero, so make your player have a value of 1 so that you don't need to change team value for every enemy.

- Is Aggressive: Used by the AI to know if the actor is wielding weapons and intends to fight or not.

Civilian Brain

Simulates a peaceful civilian. Notices enemies and follows them. Calls police.

- Only Call When Armed: The civilian will only call for police when the enemy has any weapon in hands, otherwise they will only follow and film.

- Call Chance: Chance that the civilian will make a call.

- Call Delay: Time in seconds the civilian will stay in the current state before making a call. For example, they will notice an enemy, wait for 5 seconds and only then make a call. The decision to make a call is made right when the enemy is noticed.

- Call When Fleeing: Should the civilian make calls when running away. Otherwise they will only make calls when following.

- Flee After Call: If a call was made when following, should the civilian flee even if they weren't scared in the first place.

Fighter Brain

Simulates a fighter, takes cover, assaults, uses weapons, throws grenades, etc.

- Is Assaulting: Will the AI run towards the enemy instead of taking cover.

- Chance To Take Cover After Assaulting: If "Is Assaulting" is enabled this property is used to determine if the AI will take cover after finishing an assault.

- Avoid Distance: If distance to the enemy is less than this property the AI will start avoiding them.

- Approach Distance: If AI starts an slow approach (not an investigation) they will walk towards the enemy till the distance is equal to this property.

- Assault Distance: If AI is assaulting, they will run towards the enemy till the distance is equal to this property.

- Covered Investigation Wait: Time in seconds to wait before the AI goes to inspect the position the enemy was last seen at. Used when the AI is in cover.

- Uncovered Investigation Wait: Same as covered investigation wait but applied when the AI is not in cover.

- Retreat Health: Health value at which the AI will retreat to a cover and hide.

- Hide Duration: Time in seconds the AI will stay in a cover after retreating to it.

- Stand Duration: When AI notices an enemy they will stand for a while in one spot before switching to an assault, taking cover or any other state. This property determines the time in seconds that delay will take.

- Cover Switch Wait: Time in seconds the AI will stay in their current cover before looking for another one.

- Follow Distance: If AI sees a walking uncovered enemy dissapear (for example, the enemy walks behind a wall or runs away) they will follow. They will only follow if distance to the enemy is GREATER than this property and ignores it for closer enemies.

- Grenade Reaction Time: Time in seconds after the AI notices a nearby grenade they will start avoiding it.

- Grenade Avoid Time: Time in seconds the AI will keep running away from the grenade before taking any other state.

- Debug Threat: Enables a line shown in the scene editor that is drawn between the AI and it's enemy.

- Start: Settings for the AI startup.

. Mode: How the AI starts it's existance. By default it patrols but can also immediately start searching or investigating. SearchAround and Investigate start at the gyuven position.

. Position: Position the AI will investigate or search around. Ignored for SearchAround, Idle and Patrol modes.

- Grenade: Settings for the grenades.

. Grenade Count: the amount of grenades the AI can throw.

. First Grenade Delay: AI will start considering throwing a grenade after this delay. The delay is counted after the AI becomes alerted.

. Interval: After throwing a grenade the AI will wait for this duration before considering throwing again.

. Check Interval: When considering a throw, the AI will check the situation. If the situation is unsuitable the AI will wait for this time in seconds before checking again. This parameter is used to help with the performance or make the AI throw less grenades.

. Max Radius: AI will only throw if a predicated landing position is closer to the enemy than this value.

. Avoid Distance: A grenade will be thrown if it lands further from the AI than this value. Used to avoid damaging the fighter itself.

AI Aim

Manages head, body and arm direction for both civilians and fighters.

- Speed: Speed at which the AI turns toward new directions. Can be used to make the AI aim worse at a running enemy.

- Slow Speed: When investigating or searching the AI will use this speed for turns instead of the normal one. Makes the search look smoother.

- Target Radius: Settings for the AI accuracy. AI randomly aims at a point around a target position. The point is found inside a sphere surrounding the target.

. Min: Minimum possible radius for the target sphere. Used when the enemy is closer than Min Distance.

. Max: Maximum possible radius for the target sphere. Used when the eneemy is further away than Max Distance.

Radius is interpolated between Min and Max values when the enemy is between min and max distances.

- Debug Aim: Enables debug rays inside the scene editor.

AI Backup Call

Continuously checks the situation. If the AI is in danger makes a call for backup.

- Call: Can also be used for things other than friend spawning, the function depends on the value inside the Message property.

. Target: Target object that will receive the message.

(These parameters are useful for script writers)

. Message: Function name in a script that belongs in the target object.
. Pass Caller: Should the calling Actor component be passed to the called function as an argument.

- Only Call In Cover: If enabled the AI will only make backup calls when safe in a cover.

- Trigger Count: A call is triggered if the number of nearby friends is equal or lower than this value.

- First Trigger Delay: Time in seconds counted after the AI becomes alerted to wait before the trigger is checked.

- Trigger Wait: Time in seconds to keep checking the trigger before making the call. If any friend comes closer and increases the nearby friend count during this interval the trigger is cancelled.

- Trigger Spacing: After the checking period the AI will for this period of time before going to check again.

- Max Level Count: The AI will only check if the number of friendly and aggressive AIs in the level is less than this number.

AI Communication

Finds nearby friends and makes connections to them. Other components then pass information about the threat to these nearby friends.

- Distance: Maximum allowed distance to friends.

- Update Delay: Time in seconds before the list of nearby friends will be updated again.

- Debug Friends: Draws lines inside the scene editor to each friend.

AI Cover

Finds covers when asked by the brains.

- Max Low Cover Angle: A low cover will be deemed unsuitable if the angle between it and the enemy is greater than this value.

- Max Tall Cover Angle: Same as above but for the tall covers.

- Max Cover Distance: AI will only consider covers for taking if they are closer than this value.

- Min Switch Distance: If asked to find a better cover, covers closer than this distance are ignored.

- Avoid Distance: AI avoids taking covers that are closer to the enemy than this value.

AI Fear

Checks the situation and issues a command to the brains to become scared. Can be used by both civilians and fighters.

- Only Scared Of Armed: The AI will not regard an unarmed enemy as scary. Armed enemies are those that have weapons in their hands.

- Immediate Scare Chance: Chance that the AI will run away as soon as they see the enemy.

- Min Fight Health: If health is less than this value (used like a percentage) the AI will become scared.

- Flee After Some Time: After an enemy was witnessed the AI might still run in fear after some time. This property enables or disables this functionality.

- Min Flee Time: If above property is enabled the AI will wait at least this long before running away.

- Max Flee Time: If Flee After Some Time is enabled the time the AI will flee is calculated randomly between min and max values.

- Flee On Hostile Alerts: Will the AI run away on hearing an explosion or any other similar alert. Ignores Immediate Scare Chance.

- Flee On Seeing Military: Will the AI run away when seeing a friendly but aggressive AI (like police coming to investigate).

AI Fire

Controls the equipment of weapons and their fire. Makes the AI take a peek from a cover if they are covered.

- Auto Find Index: Is the weapon discovered automatically inside the character motor.

- Index: Weapon index to use when Auto Find Index is turned off.

- Auto Find Type: Weapon type to look for, if not present any other weapon will be picked. Used only when AutoFindIndex is enabled.

- Fire If Obstructed: Should the AI fire at the target even if it is obstructed by a wall. Covers are not regarded as obstructive since the AI aims above them, waiting for the enemy to take a peek.

- Always Aim: Should the AI be always aiming, used for debugging purposes.

- Bursts: Settings for AI fire when standing.

- Cover Bursts: Settings for AI fire when the AI is covered.

AI Flashlight

Takes commands from other components and turns a flashlight on and off depending on a situation.

AI Flee

If AI becomes scared they flee towards a flee zone, ignoring everything else.

- Avoid Distance: The AI will change the direction of their run if the enemy comes closer than this value.

AI Follow

Moves and rotates the character motor so that it faces and follows the enemy when needed.

- Avoid Distance: The AI won't come closer to the enemy than this distance.

- Follow Distance: The AI will come closer if the enemy is further away than this distance.

- Use Covers: Will the AI find a cover to look at the enemy from a safe position.

- Follow From Cover Chance: Chance that the AI will follow the enemy further away from Follow Distance. Used when the AI is in cover. If the chance fails the AI will stay in cover.

- Run To Covers: If AI wants to take cover, will they run towards it.

- Use Peeks: If AI is following without filming they will usually face the direction they are walking to. If peeks are enabled the AI will continuosly look behind them to look at the enemy.

- Peek Duration:
- Peek Delay:

    Settings for those peeks.

AI Investigation

When the fighter brains enters investigation mode it passes the command of the character motor to this component. It makes the AI walk slowly towards the position the enemy was last seen at.

- Distance: A investigation will complete if the AI will see that position and is closer than this distance.

- Field Of View: Field of view for the position visibility check. The AI can be made to look directly at the position for it to be acknowledged.

- Cover Offset: If the position is behind a cover the AI will walk towards a position next to the cover offset by this property.

AI Listener

Finds alerts surrounding the AI. Without this component the AI is deaf.

- Hearing: Each alert has a distance property at which they can be heard. Hearing is used to multiply that distance. The lesser the value the closer the AI needs to be to the alerts to hear them.

AI Movement

Takes movement commands from other components and uses pathfinding to walk the character motor towards a destination.

- Debug Destination: Draws a line inside the scene editor towards a destination.

- Debug Path: Shows the path the AI will take.

AI Phone

Allows the AI to take phonecalls and film using a phone. Mostly used by Civilian Brain and AI Follow

- Call: Same as in AI Backup Call

AI Radio

Allows the AI to use a radio to take calls. Mostly used by AI Backup Call.

AI Search

Allows the AI to search the level to find an enemy. The AI searches each cover and search zone. Searches are done by positions. For example, a cover might have 4 positions the AI has to inspect. Positions are grouped into blocks and AI always checks every position inside a block before moving onto another block.

- Block Threshold: Search points belong in the same search block if they are closer to each other than this distance.

- Block Center Threshold: When constructing blocks a point is taken to be inside a block if it is closer to the center of it than this distance.

- Max Distance: AI will not search for positions that are further away than this distance.

- Verify Distance: The AI has to be close enough to a position for it to be verified.

- Verify Height: Height above ground that the AI has to see.

- Cover Offset: When a position is behind a cover the AI will walk towards a position next to the cover offset by this distance.

- Field Of View: AI will only verify positions if they fall inside this field of view.

- Walk Distance: AI will run towards positions until it comes closer than this distance.

- Max Investigation Age: Time in seconds for the AI to think that the position is safe. Used when communicating with other AIs so that they don't check it themselves.

- Debug Target: Draws a line inside the scene editor towards a search position the AI is currently going to check.

- Debug Points: Draws debug information about all known search positions.

AI Sight

Continuously updates a list of visible actors. Without this component the AI is blind.

- Distance: AI doesn't see actors that are too far away.

- Field Of View: How wide is the vision.

- Update Delay: Time in seconds to wait before updating the list of visible actors again.

- Debug FOV: Enables an arc that shows the field of view inside the scene editor.

AI Waypoints

When asked to, walks the character motor around the waypoints. The AI can be made to run or wait at each position.

Third Person Camera

Manages the camera object by setting an appropriate orientation depending on the target object's state. For camera to work you have to link it to the target object that has a Character Motor attached.

The camera component also maintains and draws a crosshair. It hides the crosshair if the character is unarmed or unable to fire at a wall because is too close. The visibility of

crosshair also can be turned off manually by setting Is Crosshair Enabled value to false when your game needs so.

Mobile Camera

Manages the camera object by setting an appropriate position depending on the target object's state. For camera to work you have to link it to the target object that has a Character Motor attached.

Camera shifts a bit depending on the orientation of the character. It also zooms out when enemies are close by in order to display a wider combat area.

Character Health

Manages health and sets Is Alive in Character Motor to false when it reaches 0. Registers damage done by bullets.

Multiple hitboxes can be setup inside the character. On setup Character Health will stop registering hits and instead will expect child Body Part Health components to pass taken damage to it.

Character Platform

Keeps character on top of a moving platform. It doesn’t require a Character Motor or any other component and therefore can be used on any object even if it’s not physical.

Additional character components

Body Part Health

Acts similarly to Character Health, but passed the taken damage to a first found Character Health component in the hierarchy.

Character Face

Reacts to character events and modifies blend shapes in a mesh. The set of faces can be expanded by modifying the script or making a new version of it.

Character Sounds

Spawns sound instances on various character events. Sounds are randomly picked from lists.

Character Effects

Spawns prefab instances on various character events.

Character Alerts

Generates alerts for the AI to pick up on various character events.

Character Name

Contains a character name used by the UI. If no Character Name is present the name of the game object is taken as the name.

Character Sleep

Turns off some components of character object when it is far away from the player's character. Components are turned on when the player approaches the character.

Additional gun components

Gun Effects

Spawns effects prefabs on various gun events like reloads or gunfire.

Gun Sounds

Generates randomised sounds on various gun events.

Gun Alerts

Generates alerts for the AI to pick up on various gun events.

Hit Effect

Spawns an effect upon a bullet hit. Used mostly on static level geometry.

Additional grenade components

Path Preview

Builds a path mesh that depicts the approximated flight of a grenade. Manipulated by player controllers.

Explosion Preview

Displays a sphere depicting an area of grenade explosion. Used by both grenades and player controllers.

Extra

Damage Trigger

Deals damage to Character Health components attached to objects that enter its trigger area.

Enemy Sight

Generates and maintains a mesh depicting field of view for a character motor.

Player Arm Trigger

Causes player characters to pick a weapon when they enter the trigger area.

Player Disarm Trigger

Causes player characters to hide their weapon upon entering the area.

Random Audio

Picks a random audio sample from the supplied list and sets Audio Source to play it.

Delayed Destroy

Disables an object after a certain amount of time passes.

Delayed Disable

Destroys an object after a certain amount of time passes.

Exit to Escape

Exits the game when an escape key is pressed. Best used in the Unity Editor.

UI

Health Bar

Takes a relative health from a Character Health component and displays a bar that displays the value.

Enemy Display Manager

Creates and manages objects with Health Bar for all enemies visible on screen.

Enemy Health

Takes a target of a character motor and sets a sibling Health Bar component to take its values.

Ammo Bar

Displays information about specific weapons ammunition. When pressed makes a character pick that weapon.

Crouch Touch

When pressed toggles associated characters crouch state.

Grenade Bar

On touch makes the character pick up a grenade.

Touch Aiming

Takes directional input from the touch screen and passes it to Mobile Controller as aim direction.

Touch Movement

Takes directional movement input from the touch screen and passes it to Mobile Controller.

Support

Ask questions, send ideas, report bugs: eduardas.ninja@gmail.com

Thanks,

RedBee Team