

**ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA**  
**CAMPUS DI CESENA**  
**SCUOLA DI SCIENZE**  
**CORSO DI LAUREA IN SCIENZE E TECNOLOGIE INFORMATICHE**

**Visita virtuale alla Rocca di Forlimpopoli  
mediante l'uso di Maya e Unity 3D**

**Relazione finale in Metodi Numerici per la Grafica**

**Relatore**  
**Lazzaro Damiana**

**Presentata da**  
**Benini Elena**

**Sessione II**  
**Anno Accademico 2014-15**



## INDICE

<b>Introduzione</b>	<b>pag. 5</b>
<b>Capitolo 1 - Computer graphics, arte e storia</b>	<b>pag. 7</b>
1.1 Introduzione	
1.2 La computer graphics applicata alla storia	
<b>Capitolo 2 - Contesto storico: la rocca di Forlimpopoli</b>	<b>pag. 15</b>
2.1 Storia della rocca	
2.2 Scelta del periodo storico da modellare	
<b>Capitolo 3 - Tecnologie utilizzate</b>	<b>pag. 25</b>
3.1 Autodesk Maya	
3.2 Unity 3D	
3.3 Altri programmi	
<b>Capitolo 4 - Realizzazione di un modello 3D della rocca di Forlimpopoli</b>	<b>pag. 27</b>
4.1 Fase preliminare	
4.2 Modellazione	
4.3 Texturing	
4.4 Semplificazione e pulizia del modello	

## **Capitolo 5 - Realizzazione della visita virtuale**

**pag. 43**

5.1 Passaggio a Unity

5.2 Lighting

5.3 Fisica e apertura porte

5.4 Movimento

*5.4.1 Spostamento*

*5.4.2 Rotazione (yaw)*

*5.4.3 Rotazione (roll)*

5.5 Creazione dell'eseguibile

## **Bibliografia e Sitografia**

**pag. 57**

Libri

Riviste

Siti

Altro

Immagini

## INTRODUZIONE

Il lavoro svolto in questa tesi mira alla creazione di un modello 3D della Rocca di Forlimpopoli mediante l'uso dei programmi Maya e Unity 3D. La combinazione di queste due tecnologie permette all'utente di utilizzare il prodotto finale per visitare virtualmente la rocca, muovendosi a proprio piacimento all'interno del modello del monumento.

Questa ricostruzione è un esempio di applicazione della computer graphics a discipline non informatiche, in questo caso specifico l'arte e la storia. Pur non sfruttando che una piccola parte delle potenzialità di tali lavori, la ricostruzione in 3D della rocca rende possibile esplorarne anche le porzioni normalmente non accessibili al pubblico, o visitabili solo in determinate occasioni.

Basandosi sulle varie fasi del lavoro svolto, la tesi è divisa in cinque capitoli, che partono dal contesto generale e procedono verso le specifiche del progetto.

Il primo capitolo presenta il contesto in cui si inserisce la tesi, cioè tra le applicazioni della computer graphics. Per descriverlo si fa inoltre riferimento ad una serie di esempi da cui è stata tratta ispirazione nella realizzazione di questo lavoro.

Il secondo capitolo è dedicato alla raccolta di informazioni sul contesto storico del monumento per valutare se sia possibile modellare l'aspetto della rocca in un periodo passato. Anche se questo non si è poi rivelato attuabile, la ricerca è stata comunque utile per familiarizzare con l'aspetto e la struttura del monumento.

Il terzo capitolo contiene una breve analisi delle tecnologie impiegate nel progetto, specificandone inoltre i vari usi al suo interno.

Il quarto capitolo descrive il lavoro di modellazione e texturing, principalmente effettuato all'interno del programma Maya, nonché la preparazione e gli accorgimenti adottati per importare il modello in Unity.

Il quinto capitolo prosegue descrivendo i passi successivi all'interno di Unity. Si inizia con il sistemare tutti gli elementi che sono stati alterati in qualche modo dall'importazione. Si prosegue, poi, descrivendo il processo di lighting della scena,

nonché le varie operazioni che hanno consentito di trasformare il modello statico in uno interattivo.

# CAPITOLO 1

## COMPUTER GRAPHICS, ARTE E STORIA

### 1.1 Introduzione

Lo sviluppo della computer graphics o CG è concomitante a quella del progresso dei computer stessi. Da quando sono state disponibili le prime macchine, in diversi ambiti si è presentata la necessità di trasformare e di rappresentare visualmente le informazioni numeriche da esse prodotte o elaborate, passando quindi da un modello matematico ad un'immagine.

La computer graphics è dunque il settore dell'informatica che studia le tecniche e gli algoritmi che consentono la creazione, la modifica e la rappresentazione visuale di oggetti reali o astratti.

Già dalla definizione è evidente che le applicazioni di questa disciplina sono molteplici e di diffusione capillare. Basti pensare alle Graphical User Interface o GUI, presenti praticamente in tutte le applicazioni per ambienti personal computer e per workstation, che sono alla base della diffusione dell'uso di queste macchine da parte di utenti di qualunque livello.

Tuttavia l'utilità della computer graphics va al di là della stessa informatica, intrecciandosi con le altre scienze, con l'architettura e con l'arte [Dre13; SCMS05].

Tra gli esempi più noti vi sono le sue applicazioni all'interno dell'industria dell'intrattenimento. Animazione al computer ed videogiochi sono due tra i principali settori che hanno spronato la ricerca e lo sviluppo nel campo della CG, seppure in direzioni differenti. Entrambi sono partiti come contributi per illustrare ricerche scientifiche: l'animazione come visualizzazione scientifica [Car03; Sev12], e i videogiochi come mezzo per dimostrare le performance di un computer ad un pubblico di non esperti [Car03; Dre13] o come lavoro di interesse teorico nella speranza che la soluzione trovata fosse applicabile a problemi analoghi [Dre13].

In effetti, ciò che è stato sviluppato risulta essere un potente strumento e, superati i problemi tecnici, il focus si sposta su considerazioni di natura più artistica, come la ricerca di elementi iconici e memorabili e il più alto livello di fotorealismo raggiungibile [Mor14.1; Mor14.2; NG13].

Se quest'ultimo si rivela essere relativamente facile per l'animazione, tanto che già negli anni Novanta iniziavano ad esserci scene di CG così realistiche da risultare difficilmente riconoscibili anche ad un occhio esperto [Sev12], i videogiochi sono da sempre più limitati per la loro necessità di effettuare il rendering delle scene in tempo reale. Tuttavia, dal punto di vista visivo questi hanno sicuramente un vantaggio rispetto all'arte tradizionale, cioè la possibilità da parte del giocatore di interagire con un mondo virtuale. Se non si considera questa interazione, ciò che rimane è una raffigurazione statica, il cui successo si basa sugli stessi canoni dell'arte tradizionale [Sol12].

Si può quindi definire la computer graphics un nuovo ramo nella storia dell'arte. Infatti, sia che si disegni su carta o tela, sia che si disegni sullo schermo di un computer, si tratta sempre di una superficie piatta e inanimata, un qualcosa che di per sé non ha profondità, da trasformare in un mondo vivente con l'illusione di una terza dimensione. L'unica differenza è che, specie nella computer graphics 3D, il computer automatizza molte difficoltà tecniche del calcolare la prospettiva e le luci della scena [Sol12].

Non solo la computer graphics è una forma di arte, ma quest'ultima è stata in molti casi il motore che ha spinto ad esplorare e creare nuove possibilità di utilizzare un computer. Il percorso di evoluzione della computer graphics è complesso e intricato, con scoperte fatte in un ambito che vengono riproposte e ulteriormente sviluppate in un altro. Inoltre, col proseguire della ricerca, i suoi usi sono diventati sempre più vari.

## 1.2 La computer graphics applicata alla storia

Le tecniche 3D possono essere messe al servizio della ricerca storica, come strumenti sia di diffusione ad un largo pubblico, sfruttando anche le molteplici possibilità offerte dal Web, sia al servizio del processo interpretativo o di restauro.

Uno degli elementi che la rende eccellente come strumento divulgativo è il fatto che la modellazione 3D permette di ricreare l'aspetto di luoghi e oggetti inaccessibili per



**Figura 1: Ricostruzione del tempio di Phimai, Thailandia.**

svariati motivi. Tra di essi si spazia dalla semplice distanza geografica a esigenze di conservazione che impongono di limitare il numero di visitatori [Lev01; Spa07; Rem10], alla possibilità di reinserire oggetti e reperti conservati per ragioni di sicurezza all'interno di musei posti nei dei luoghi dove sono stati trovati [Min13]. O ancora, tramite la creazione di un museo

virtuale, permettono di 'esporre' al pubblico reperti conservati in locali di deposito per mancanza di spazi espositivi o di riunire in un unico luogo collezioni disseminate in svariate sedi [TAB10]. In tutti questi casi è utile potersi affidare ad un intermediario, come può essere una riproduzione digitale, che può essere ulteriormente arricchita con informazioni e spiegazioni fornite all'utente nel corso della visita virtuale [CNE11].

A livello di ricerca, effettuare un Restauro Virtuale può essere utile per sperimentare operazioni di restauro di un reperto o di una architettura proponendo diverse soluzioni e valutando il risultato finale prima di andare ad agire sull'oggetto stesso, senza quindi rischiare di danneggiarlo. Inoltre, qualora un restauro tradizionale non sia possibile, il Restauro Virtuale fornisce una valida alternativa. Rende infatti possibile ripristinare aree dove eventuali pitture siano sbiadite o abrase, nonché ricostruire estensivamente parti mancanti seguendo anche ipotesi

arbitrarie o di natura artistica [DJL15], operazioni che nel restauro tradizionale vengono evitate. Il Restauro Virtuale evita il problema in quanto non agisce direttamente sul reperto ed il risultato può essere facilmente modificato qualora nuove scoperte rendano invalida la ricostruzione precedente. Per questi stessi motivi viene anche ignorata la convenzione del restauro tradizionale secondo la quale la parte ricostruita deve essere ben distinguibile dall'originale [Lor12].

Altro uso della ricostruzione virtuale può essere verificare ipotesi e facilitare quindi il processo di interpretazione dei reperti archeologici. Estremamente esemplificativo è il caso di un sito minoico in Grecia [PK10] all'interno del quale è stata ritrovata una stanza contenente strumenti per la realizzazione di oggetti in terracotta. L'ipotesi che si trattasse di un laboratorio per la produzione di ceramiche è stata smentita modellando l'ambiente in 3D e indagandone l'illuminazione, rivelatasi insufficiente in ognuna delle ipotesi ricostruttive; il tutto per mezzo di uno studio che si sarebbe rivelato di difficile attuazione con altri mezzi.

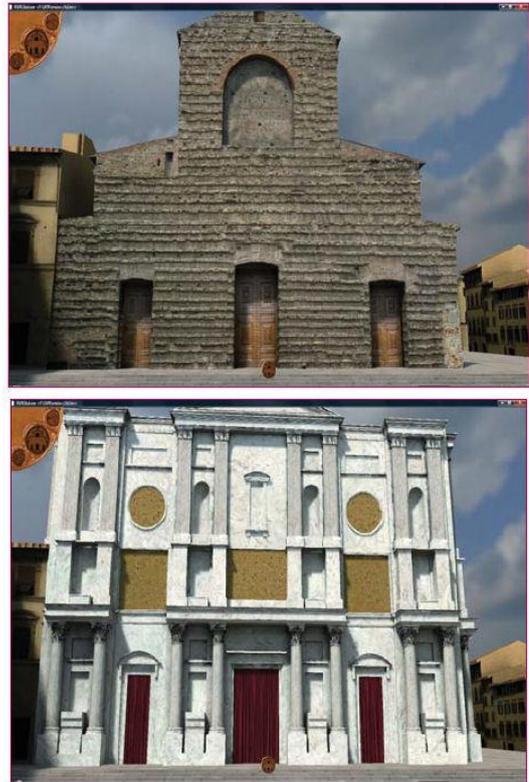


**Figura 2: Ricostruzioni del laboratorio di ceramiche con diverse illuminazioni.**

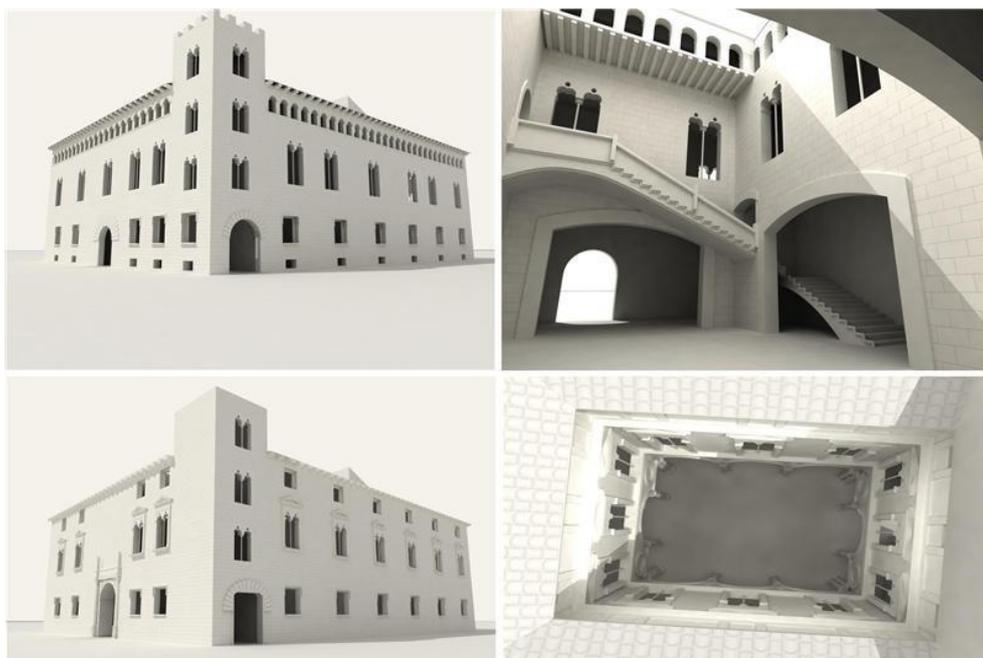
A tutto ciò si aggiunge la possibilità, specialmente nel caso di un edificio o di un'area, di ricostruire all'interno di una simulazione le trasformazioni che si sono susseguite nel corso della storia in quel determinato luogo [Spa07]. O ancora, di creare per la prima volta in digitale opere progettate e mai portate a termine, o

anche completamente immaginarie, come le città ideali immaginate da svariati autori del passato [CNE11].

Le applicazioni sono dunque molteplici e difficilmente esauribili in un'unica descrizione, ma la realizzazione comporta una mole di lavoro non indifferente. Ancora prima di iniziare occorre infatti effettuare una ricerca archeologica storica e letteraria per ottenere informazioni il più dettagliate possibile sul luogo o l'oggetto che si desidera ricreare. In particolare sono indispensabili: dimensioni, materiali ed altre informazioni specifiche come nel caso di edifici, piante e prospetti. Eventuali elementi mancanti possono essere dedotti da altri reperti o edifici dello stesso periodo e località [Her13; Lev01; Spa07].



**Figura 3: Due diverse ricostruzioni della chiesa di S. Lorenzo a Firenze. In alto nel suo aspetto reale, in basso secondo il progetto michelangiolesco.**



**Figura 4: Palazzo dell'Ambasciatore Vich a Valencia, distrutto nel XIX secolo e ricreato virtualmente dai ricercatori delle università di Murcia e Valencia nel 2013.**

Una volta raccolte queste informazioni occorre effettuare una selezione critica dei dati per stabilire il livello di dettaglio della ricostruzione, basandosi soprattutto sul rapporto tra la scala e i contenuti del modello, nonché lo scopo ultimo per cui viene realizzato. L'obiettivo è evitare di aggiungere durante la fase di modellazione dettagli insignificanti il cui unico effetto è aumentare le dimensioni del file su cui si sta lavorando. Un esempio banale di questo è la creazione di un muro con una sola primitiva piuttosto che riproducendo separatamente ogni singolo mattone.

Solo a questo punto inizia il lavoro al computer, utilizzando uno o più programmi specifici a seconda del tipo di ricostruzione che si vuole effettuare [Spa07]. Si può infatti realizzare un'animazione che guidi il "visitatore" lungo un percorso prefissato, oppure creare uno spazio interattivo molto simile ad un videogioco e lasciare all'utente libertà di movimento al suo interno.

Viste queste premesse, il lavoro di questa tesi è iniziato con una fase di ricerca e raccolta dati, volta a decidere se modellare la Rocca di Forlimpopoli nella sua forma attuale o se proporre una versione passata non più visitabile al giorno d'oggi, propendendo in entrambi i casi per una ricostruzione di tipo interattivo.

Di seguito alcuni rendering a confronto con le fotografie scattate sul posto.



**Figura 5: Foto.**



**Figura 6: Rendering realizzato con MentalRay.**



**Figura 7: Foto.**



**Figura 8: Rendering realizzato con MentalRay.**

## CAPITOLO 2

### CONTESTO STORICO: LA ROCCA DI FORLIMPOPOLI

#### 2.1 Storia della rocca

Un primo fortilizio, indicato con il nome *Salvaterra*, fu costruito tra il 1361 e il 1364 per ordine del Cardinale Albornoz, dopo che questi aveva conquistato e distrutto la città di Forlimpopoli ponendo fine all'opposizione degli Ordelaffi (e in particolare di Francesco II) all'autorità papale. La rocca serviva al governo del Papa per il controllo militare e fiscale della zona ed il sito scelto per l'edificazione fu quello della precedente cattedrale romanica di S. Maria Pupiliense, demolita fino alle fondamenta per completare la nuova costruzione [Ald99; Bar14; Ber70; CG05; Ted92].

Dopo la morte in esilio di Francesco II, suo figlio Sinibaldo Ordelaffi riuscì a porre fine alle ostilità con la Chiesa a cominciare dal 1379 ed ottenne il permesso di ricostruire la cinta muraria della città, permettendo agli abitanti di tornare. Storici passati gli hanno attribuito anche lavori di ampliamento della rocca, ma ricerche recenti hanno eliminato questa possibilità [Ald99]. Rimane il fatto che da questo errore deriva la duplice denominazione di Rocca Albornoziana e Rocca Ordelaffiana.

La fortezza è costruita in mattoni e calcestruzzo su pianta trapezoidale. Originariamente aveva agli angoli alte torri a pianta quadrangolare analoga a quella del possente bastione che si erge sul lato meridionale a difesa dell'ingresso e del ponte levatoio. Al centro delle mura sorgeva il mastio, un'ampia torre centrale quadrata di 14 x 14 m alla base più alta di quelle poste agli angoli [Ald99; Ted92].

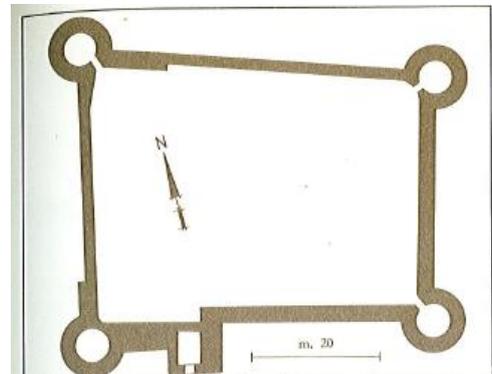


Figura 9: Pianta della rocca di Forlimpopoli (rilievo di F. Montevicchi).

La rocca ospitò molte famiglie che si alternarono frequentemente al potere. Infatti, tra il 1400 e al 1500 Forlimpopoli passò ripetutamente di mano in mano. Alcuni tra i contendenti furono la Chiesa, gli Sforza, i Riario e gli stessi Ordelaffi.

Furono questi ultimi a provvedere ad un primo restauro della rocca tra il 1471 e il 1480 per ordine di Pino III. Altri lavori furono effettuati poco dopo per ordine di Caterina Sforza. In entrambi i casi si trattò di un rafforzamento per mantenere le difese efficienti nonostante il progresso bellico, in particolare per rendere la rocca più resistente al tiro delle artiglierie.

A questo scopo le cortine furono ingrossate, munendole in basso di scarpe, cioè di tratti di muro inclinati. I bastioni d'angolo furono abbassati al livello delle cortine e raccordati da camminamenti coperti. È in questo secolo che queste torri persero inoltre la loro originaria pianta poligonale per assumere l'odierna forma circolare. In

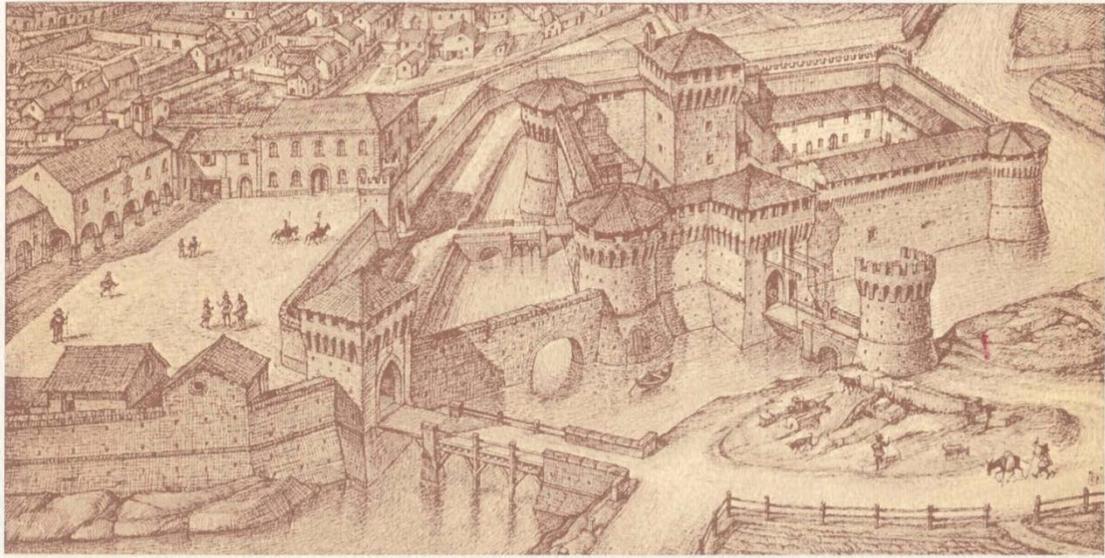


**Figura 10: Ghiacciaia alla base della torre nord-est, nella Sala II del Museo Archeologico Tobia Aldini.**

seguito subirono delle ulteriori modifiche, questa volta alla base, portando alla creazione di ghiacciaie [Ald99; Ber70; Ver07].

Inoltre, la rocca era circondata da un fossato pieno d'acqua profondo sei metri e largo dieci nel quale l'Ausa versava le sue acque e che si congiungeva al fossato circostante le mura cittadine. La rocca era infatti isolata all'esterno dell'abitato, a cui era collegata

tramite uno stretto ponte levatoio sul lato est. Del rivellino eretto a rinforzo dell'ingresso principale alla rocca nel bastione meridionale oggi non rimane più traccia, come di buona parte del fossato. Ciò nonostante, si può dire che in questo secolo la rocca assunse l'aspetto che in linea di massima conserva ancora oggi [Ald99].



**Figura 11: Rocca di Forlimpopoli fra i secoli XVI-XVII. Ricostruzione di P. Novaga. Pro Loco Forlimpopoli - "Un dèint la Ròca ad Frampùl" XVII edizione - 1995**

La città continuò ad essere contesa, passando sotto il dominio dei Borgia, brevemente di nuovo degli Ordelaffi, altri inviati papali, e degli Zampeschi (1535). Tra questi ultimi merita di essere citato Brunoro II, sotto il cui vicariato il Rinascimento

giunse finalmente alla corte di Forlimpopoli, facendole vivere il suo momento di massimo splendore. Tracce di questo periodo sono visibili grazie a tracce di decorazioni rinascimentali nei soffitti a volta del primo piano [Ald99; Ted92].



**Figura 12: Dettaglio delle decorazioni rinascimentali del soffitto del primo piano.**

Con Brunoro II si estinsero gli Zampeschi e Forlimpopoli tornò sotto il dominio diretto della chiesa. Alla vedova di Brunoro, Battistina Savelli, il Papa concesse di continuare a risiedere nella rocca, ma dopo la sua morte nel 1591, concesse la rocca di Forlimpopoli in enfiteusi, cioè mediante un contratto d'affitto, a partire dall'anno successivo al nipote di Battistina, il principe Paolo Savelli di Roma. Egli mantenne la rocca come sua residenza estiva fino al 1621, anno in cui vendette l'intero capitale ereditato dalla zia al cardiale Luigi Capponi, arcivescovo di Ravenna. Il Papa a sua volta concesse la rocca in enfiteusi ai nuovi proprietari, cioè

ai marchesi Capponi di Firenze. Da residenza estiva la rocca venne ulteriormente declassata a magazzino di raccolta dei prodotti agricoli provenienti dai loro numerosi possedimenti terrieri in zona [Ald99; Bar14; Ver07].

Ormai venuta meno la funzione militare della fortezza, iniziò la sua lenta decadenza.

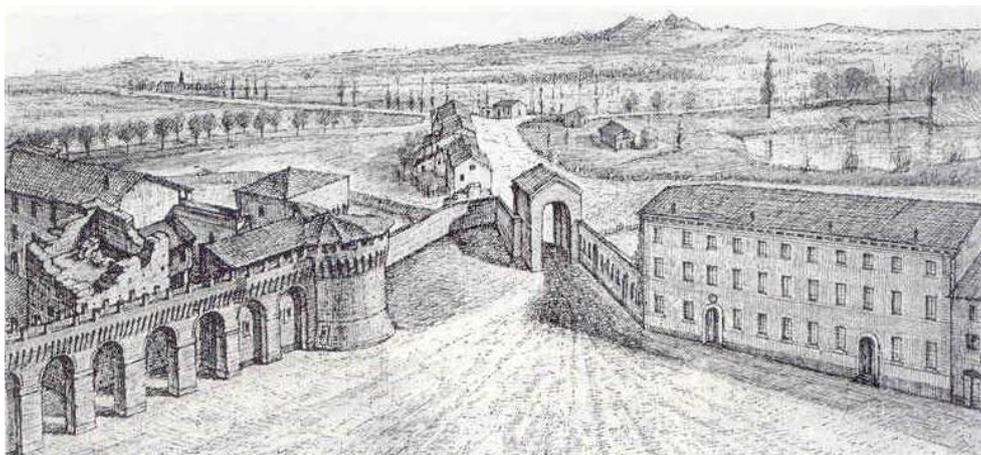
Tra il 1773 ed il 1783 opere urbanistiche volte a realizzare spazi pubblici più idonei alle nuove esigenze dei cittadini cambiarono profondamente l'aspetto della zona in cui sorgeva la rocca. Il tratto di mura che girava attorno alla fortezza sui lati nord e ovest venne abbattuto e nuovi tratti agganciarono le mura alla rocca, in particolare al torrione nord-est dell'edificio e alla torre di guardia sovrastante il ponte levatoio a sud.

La porzione di fossato che divideva la fortezza dall'abitato venne riempita ed il ponte levatoio a ovest distrutto. La piazza principale, l'odierna piazza Garibaldi, risultò notevolmente ampliata da questi lavori e l'attuale Piazza Pompilio fu creata ex-novo.

Ulteriori lavori comprendevano la demolizione del mastio per ampliare il cortile interno e recuperare mattoni da utilizzare per la costruzione di nuove case cittadine, e la realizzazione delle aperture ad arco semicircolare nel bastione meridionale per sostituire l'antico ingresso ormai murato [Ald99].

Durante la temporanea dominazione napoleonica il Papa firmò il Trattato del Tolentino nel 1797 che sanciva la rinuncia della sovranità del pontefice sui territori della Chiesa occupati dai Francesi. Di conseguenza il marchese Capponi fu privato della rocca che passò alla comunità locale rimanendo tale anche dopo il ritorno sotto al Papa come sancito dal Congresso di Vienna 1814-1815. Nel frattempo la rocca era divenuta la sede del municipio e tale restò fino al 1862 [Ald99; Bar14].

Continuava la demolizione del mastio e nel 1815 vennero aperte sette arcate (di cui tre puramente decorative) nella cortina di ponente per facilitare il libero transito dalle odierne piazza Garibaldi a piazza Fratti [Ald99; Bar14; Ted92; Ver07].



**Figura 13: Angolo sud-est di Piazza Garibaldi negli anni 1815-20. Dettaglio. È messa in risalto la demolizione del mastio e le arcate aperte nella cortina di ponente.**

Pur favorendo lo sviluppo commerciale, tutti questi interventi furono fatti senza alcun riguardo per l'integrità della rocca ed il suo valore storico, una tendenza durata per tutto l'Ottocento.

Della vecchia residenza del castellano al giorno d'oggi, infatti, restano ancora intatti pochi ambienti dato che le sale sono state più volte demolite e rifatte [Ald99].

Il lato sud in particolare fu consolidato e ristrutturato nei primi decenni



**Figura 14: Interno del teatro Verdi.**

dell'Ottocento per poi permettere nel 1830 di stabilirvi il piccolo teatro Verdi, rimasto nella storia quando nella notte del 25 gennaio 1851 il brigante Stefano Pelloni, detto il "Passator cortese", insieme alla sua banda depredò il pubblico, composto dai cittadini più benestanti della città, conducendoli alle loro case e costringendoli a consegnare denaro e preziosi. Il teatro subì poi dei lavori

tra il 1877 ed il 1882 che lo modificarono radicalmente. L'ala sud fu ampliata per poter realizzare un ingresso più elegante e funzionale, mentre all'interno fu abbassato il palcoscenico e furono costruite le due gallerie costruite da colonnine in ghisa tuttora esistenti [Ald99; CG05; Col02].

Negli anni Settanta dell'Ottocento l'ala orientale dell'edificio subì nuovi rifacimenti dando origine alla Sala del Consiglio Comunale al primo piano.

Dopo il 1862 il municipio si trasferì altrove e per la rocca seguì un secolo di abbandono e degrado, sia funzionale che strutturale [Ver07].

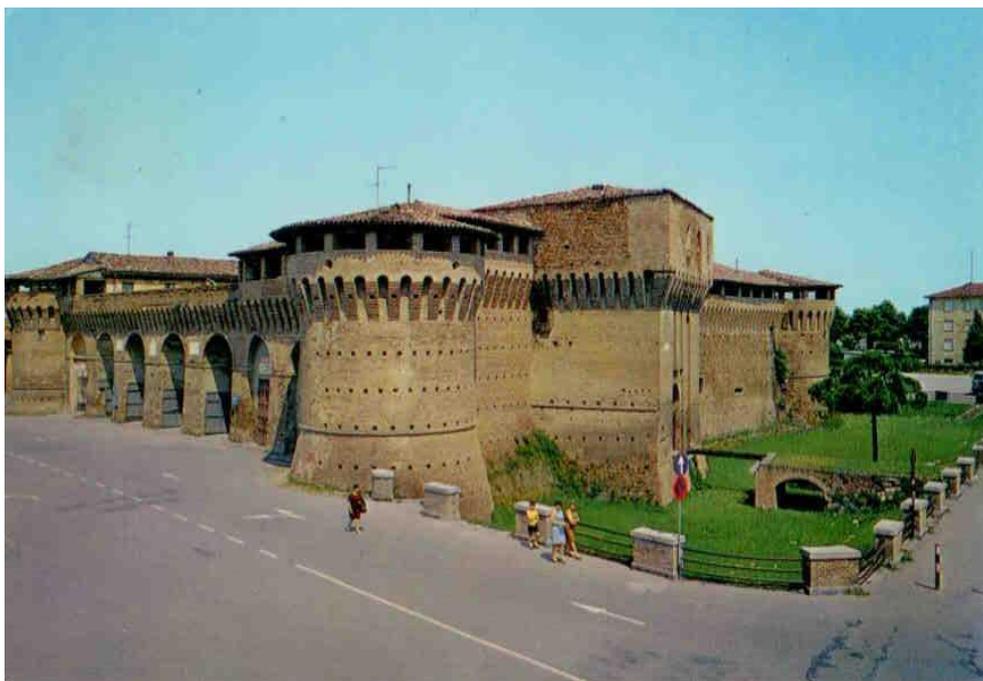


**Figura 15: La rocca nel 1902**

Verso la fine della seconda guerra mondiale la rocca venne danneggiata da mine tedesche il cui scopo era abbattere il Palazzo della Congregazione di Carità, che sorgeva nelle immediate vicinanze. Fortunatamente, la ricostruzione avvenuta verso la fine degli anni Quaranta restaurò la rocca senza ripristinare l'edificio abbattuto. Vennero inoltre intrapresi scavi considerevoli per ricreare parzialmente l'antico fossato e con esso fu ripristinato l'ingresso che in origine era a ponte levatoio [Ald99; Ber70].



**Figura 16: La Rocca danneggiata dalle mine tedesche verso la fine della Seconda Guerra Mondiale.**



**Figura 17: La rocca prima dei lavori realizzati nel 1970**

Nel 1961 fu aperto il Museo Civico Archeologico al pianterreno dell'ala orientale. I locali che lo ospitano hanno conservato meglio di ogni altro la forma originaria e contengono principalmente i reperti preistorici, romani, medievali e rinascimentali recuperati negli scavi in zona. Tra questi, i resti della cattedrale distrutta dall'Albornoz[Ald99; CG05; Col02].



**Figura 18: Interno del Museo Archeologico Tobia Aldini.**



**Figura 19: Il fossato prima dei lavori del 2007.**

Dal 1976 la rocca tornata sede del municipio e l'ufficio del sindaco venne arredato con i mobili appartenuti a Pellegrino Artusi, il cittadino storicamente più illustre della città [Col02]. In quello stesso periodo è stato realizzato il più recente intervento di restauro volto a recuperare l'antico decoro, più precisamente dal 1970 al 1990 [Bar14].

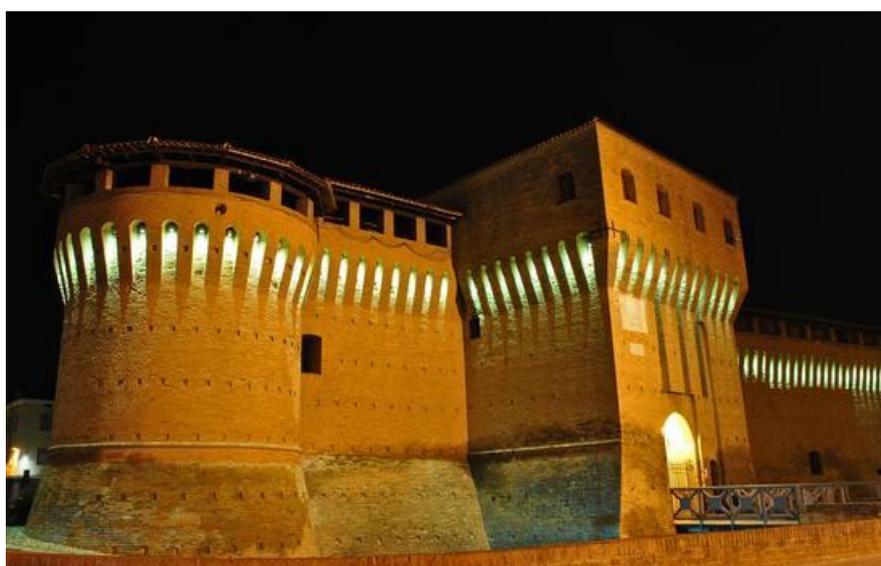
I lavori realizzati più di recente risalgono però al 2007 quando è stato creato il giardino di fronte alla rocca sul lato occidentale, memoria storica del tratto di fossato non più presente. Lo spazio verde si allunga poi verso il torrione sud e scende nel fossato, rendendolo più accessibile al pubblico e sostituendo le scale presenti in precedenza. A questa data risale anche l'illuminazione notturna della rocca, studiata per metterne in risalto i dettagli architettonici [OA09].



**Figura 20: Il fossato dopo i lavori del 2007.**



**Figura 21: Il nuovo giardino di fronte alla rocca.**



**Figura 22: Illuminazione notturna della rocca.**

A causa di queste continue modifiche una modellazione accurata della rocca in una delle passate epoche storiche risulta estremamente difficile. In particolare rimangono dubbi sulla posizione e l'ampiezza del mastio, tanto che si è preferito ricostruire la rocca nel suo presente stato, permettendo di visitare virtualmente anche quei locali che in condizioni normali non sono accessibili al pubblico. Inoltre è stata omessa la modellazione dell'interno del teatro Verdi, in quanto già utilizzato in passato per un simile progetto.

## **2.2 Scelta del periodo storico da modellare**

La storia della rocca è estremamente articolata, con numerose modifiche e rifacimenti che si sono susseguiti nei secoli e, più di recente, restauri volti a riavvicinarla alla sua forma originale. Viste le notevoli variazioni strutturali e la difficoltà nel reperire informazioni sufficientemente dettagliate per una ricostruzione fedele durante qualsivoglia secolo passato, si è preferito modellare la rocca com'è oggi, scelta incentivata anche dalla disponibilità di piante e prospetti e dalla possibilità di osservare e fotografare i dettagli non deducibili dagli stessi.

In realtà, nel corso della modellazione la struttura è cambiata nuovamente a causa del cambio di gestione del museo, che ha visto l'apertura di una nuova porta di più facile accesso verso una delle sale secondarie. Per motivi di accuratezza si è preferito modellare l'aspetto di questo particolare dettaglio prima della modifica, rimanendo fedeli alla pianta che si aveva a disposizione.

Per quanto riguarda gli arredi si è preferito dare un'idea del genere di ambienti che si incontrano all'interno della rocca, senza riprodurre esattamente i mobili e la loro disposizione nelle varie stanze. Inoltre, l'interno del teatro Verdi non è stato modellato, in quanto oggetto in passato di un lavoro simile in un'altra tesi.

Per finire, come già accennato nel capitolo precedente, si è deciso di proporre una visita interattiva, permettendo all'utente di muoversi liberamente all'interno del modello e di esplorare gli ambienti a proprio piacimento.



## CAPITOLO 3

### TECNOLOGIE UTILIZZATE

#### 3.1 Autodesk Maya

Nato come software di intrattenimento, Maya è ben presto diventato uno dei più importanti strumenti dell'industria dell'animazione [Car03].

Il suo sviluppo fu iniziato nel 1993 da Alias, a cui poi si aggiunse Wavefront nel 1995, quando le due compagnie si unirono formando Alias/Wavefront (poi acquistato da Autodesk nel 2005-06). Entrambe si erano da subito contraddistinte per il loro impegno nel creare software grafici fino a quel momento non esistenti sugli scaffali e tra il 1993 e il 1998 misero a frutto quest'esperienza portando alla creazione di Maya [Car03].

Maya fu utilizzato sin da subito in produzioni di successo ed è tutt'oggi considerato uno dei programmi leader nell'industria. Tutto ciò è dovuto anche a numerosi aggiornamenti e migliorie nel corso degli anni, volti ad incorporare le tecnologie più all'avanguardia ed ad espandere il numero di piattaforme su cui può essere utilizzato [Car03].

Per citare alcune delle produzioni che se ne sono servite, nel 2000 i quattro videogiochi per Playstation 2 più venduti erano stati realizzati con Maya. In quello stesso anno anche tutte e tre le produzioni cinematografiche nominate all'Oscar per Best Visual Effects avevano usato Maya: *The Matrix* (Mane), *Stuart Little* (Sony) e *Star Wars Episode I "The Phantom Menace"* (ILM). *The Matrix* ottenne l'Oscar [Car03].

Per quanto riguarda questa tesi, il programma è stato utilizzato per la prima parte del lavoro, dedicata principalmente alla modellazione ed al texturing della Rocca.

#### 3.2 Unity 3D

Questa compagnia fu fondata nel 2004 con l'obiettivo di "democratizzare lo sviluppo dei videogiochi", cioè creare uno strumento che rendesse semplice la creazione di

videogiochi 3D ad alta qualità anche da parte di piccoli sviluppatori indipendenti [Tak14].

In pratica raggruppò una serie di elementi di grafica 2D e 3D difficili da realizzare in pacchetti semplici e accessibili anche a chi non dispone di un software proprietario [Tak12; Bro13]. Unity 3D è dunque un insieme di strumenti che si occupa della grafica, dell'audio, della fisica, delle interazioni e del networking. Disponendo delle tecnologie alla base della creazione di videogiochi già pronte all'uso, gli sviluppatori sono liberi di concentrarsi sul processo creativo. È possibile creare giochi senza scrivere una linea di codice, ed al contempo è possibile aggiungere script per ottenere gli effetti desiderati [Bro13].

Altro elemento chiave è la possibilità di creare giochi in grado di funzionare su un vasto numero di piattaforme, tra cui sono inclusi device mobili, console, browser, PC, Mac, e persino Linux sfruttando le caratteristiche di ciascuna per funzionare al meglio [Tak14; Bro13].

Il business di Unity3D è cresciuto molto da quando è nato, sia come diffusione che come livello qualitativo. Attualmente si riescono ad ottenere demo che hanno un aspetto simile a quello di film generati al computer [Tak12].

### **3.3 Altri Programmi**

Principalmente gli altri programmi coinvolti nella realizzazione di questa tesi sono stati Paintshop Pro e Photoshop. Entrambi noti programmi di grafica, sono stati utilizzati per la creazione e la modifica delle texture a partire da fotografie o immagini. Principalmente il lavoro è stato svolto in Paint Shop Pro data la maggiore semplicità di utilizzo, sfruttando Photoshop per ottenere alcuni risultati specifici, dato che quest'ultimo ha alcune funzioni non presenti nel primo programma [Pre10].

L'ultimo programma da menzionare è ZoomBrowser Ex, un programma di utility della fotocamera. Qui se ne è utilizzato unicamente uno strumento collegato ad una funzione corrispondente della fotocamera per la composizione di fotografie adiacenti in un'unica immagine.

## CAPITOLO 4

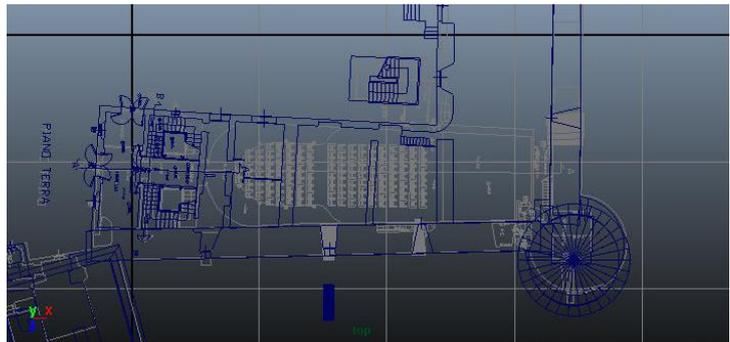
### REALIZZAZIONE DI UN MODELLO 3D DELLA ROCCA DI FORLIMPOPOLI

#### 4.1 Fase Preliminare

Come già anticipato, prima di iniziare a lavorare con Maya è stata necessario reperire presso gli uffici del Comune di Forlimpopoli i file .dwg contenenti la pianta ed i prospetti esterni della Rocca. A questi si è associata un'attenta osservazione della costruzione, anche scattando numerose fotografie, volta a meglio definire dettagli omessi o poco chiari nei disegni. Particolare attenzione è stata data al cortile interno (Piazza Fratti) in quanto non è stato possibile trovarne i prospetti.

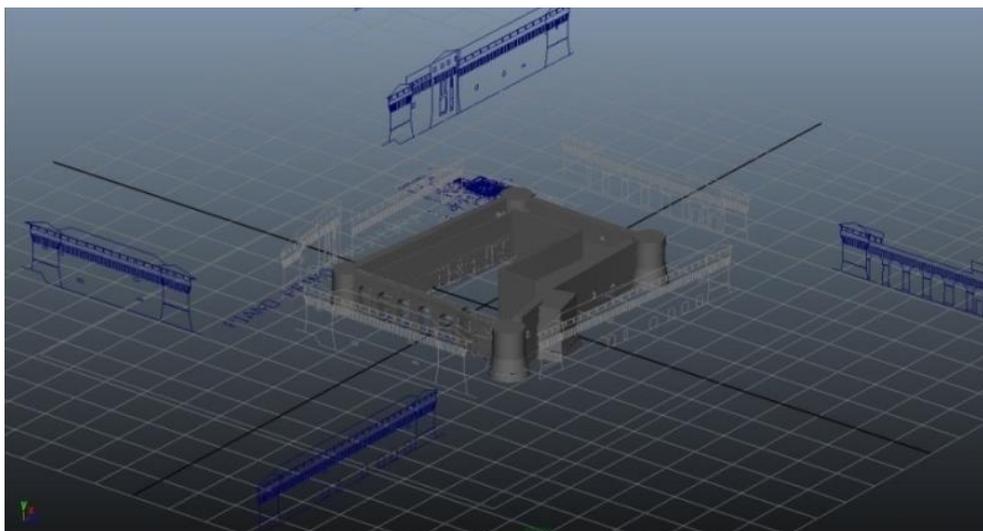
#### 4.2 Modellazione

Il contenuto dei file .dwg è stato importato in Maya e disposto in modo favorevole alla modellazione. Le piante dei vari piani sono state sovrapposte dopo averne raggruppato gli elementi in modo che fosse possibile sfruttare lo strumento *Template* per



**Figura 23: Sovrapposizione delle piante.**

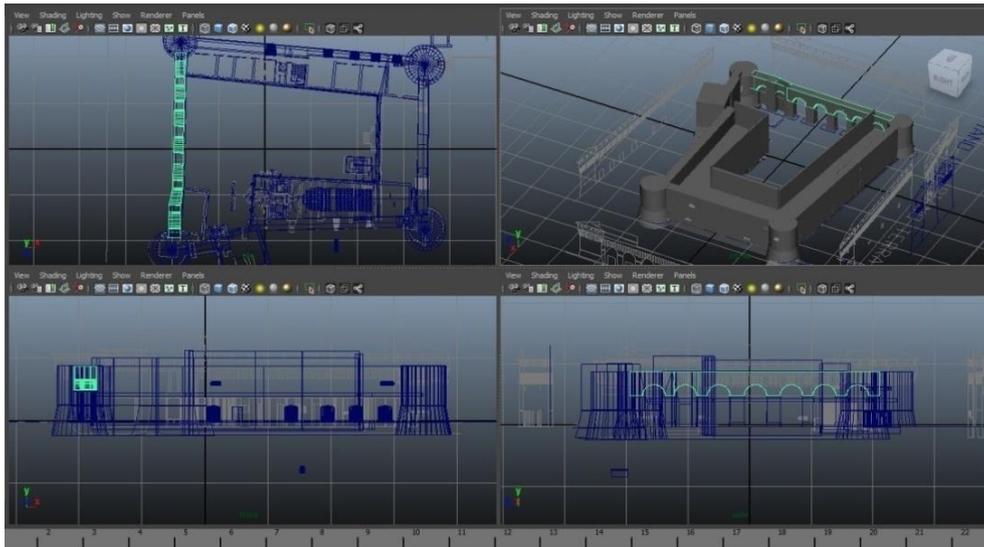
congelarle, rendendo attiva e quindi ben visibile solo quella di volta in volta necessaria al lavoro. Un lavoro analogo è stato fatto per i prospetti, per i quali si è aggiunta la necessità di tenerne due copie: una parallela agli assi per consentire una più facile modellazione degli elementi ed una con la stessa rotazione delle pareti per assicurare il corretto posizionamento delle stesse.



**Figura 24: Disposizione dei prospetti.**

Durante l'organizzazione delle piante è subito risultato evidente che, dopo una storia costellata da rifacimenti e modifiche, l'edificio è rimasto caratterizzato da una struttura molto irregolare. Ad esempio, è difficile trovare coincidenze nella disposizione dei muri interni dei diversi piani, oppure nella posizione delle finestre delle cortine. Le mura stesse al secondo piano si allargano per dare origine ai camminamenti, richiedendo due fasi di lavoro distinte. Da ciò è derivata la necessità di modellare l'interno di ogni piano singolarmente, procedendo in ordine a partire dal più basso.

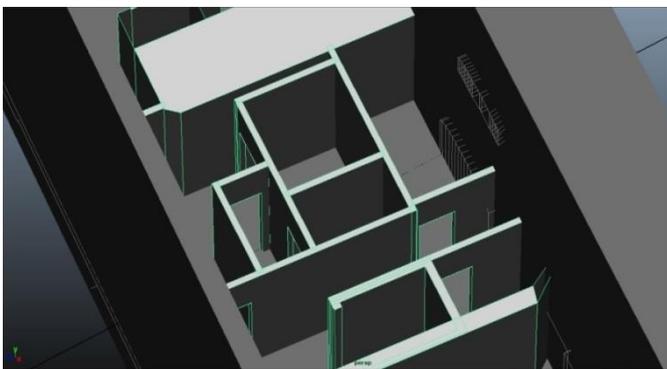
Il primo passo della modellazione è stato quello di ricreare le mura esterne, punto di riferimento fondamentale di tutto il lavoro. Per la realizzazione delle arcate della cortina di ponente si è scelto di suddividere la struttura, realizzando le colonne separatamente dalla parte superiore, ottenuta per mezzo di sottrazioni booleane tra la parete e dei cilindri opportunamente ruotati e traslati.



**Figura 25: Cortina di ponte.**

Le torri d'angolo, più larghe alla base, sono state realizzate partendo da un poligono (un cilindro, per l'esattezza) sfruttando lo strumento di estrusione *polyExtrude* ed i prospetti come guida per le proporzioni.

Anche le pareti che danno su Piazza Fratti sono state realizzate per mezzo di



**Figura 26: Muri interni del piano terra.**

estrusioni di un poligono base seguendo le piante, e lo stesso procedimento è stato utilizzato per i pavimenti.

Inoltre sono stati preparati i poligoni che in un secondo momento avrebbero dato origine alle aperture delle finestre per mezzo di sottrazioni

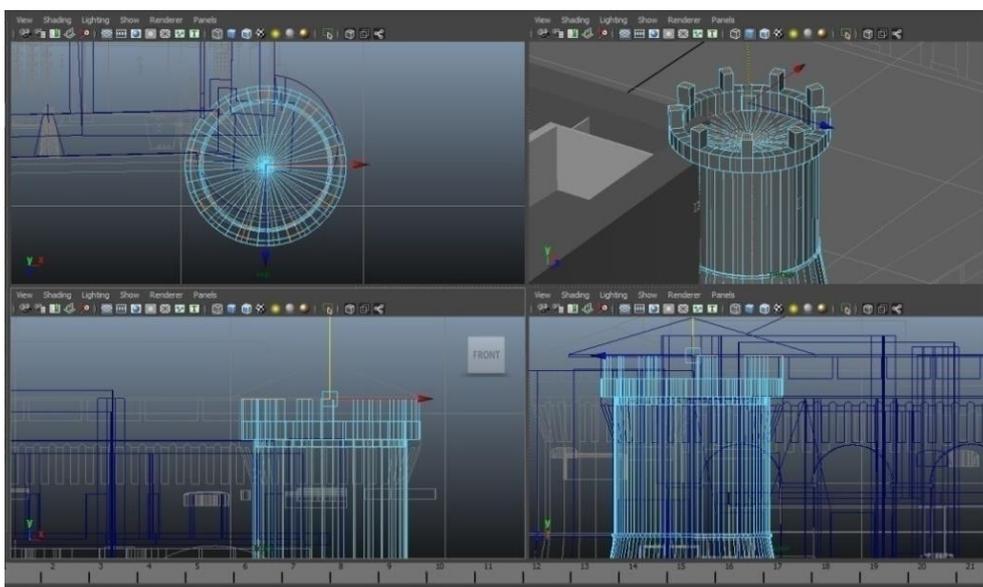
booleane. Sono poi stati messi da parte utilizzando la funzione *Template*.

Man mano che si creavano le pareti interne si procedeva alla realizzazione dei vani delle porte tramite una serie di sottrazioni booleane. L'uso di uno stesso parallelepipedo duplicato più volte e traslato in posizione prima di sottrarlo alle pareti ha consentito di rendere le aperture uniformi in altezza, mentre la larghezza è stata adattata dove opportuno.

Arrivati al secondo piano, si sono ripresi lavori lasciati a mezzo, come le mura e le torri, aggiungendo poi i tetti, prima di procedere ai dettagli dell'interno che necessitavano di tutto ciò come riferimenti.

Per le torri si è continuato con delle estrusioni, dando forma ai parapetti ed alle colonnine che sostengono il tetto. L'interno è stato poi reso cavo per poter modellare le stanze all'interno e le ghiacciaie alla base di due di esse.

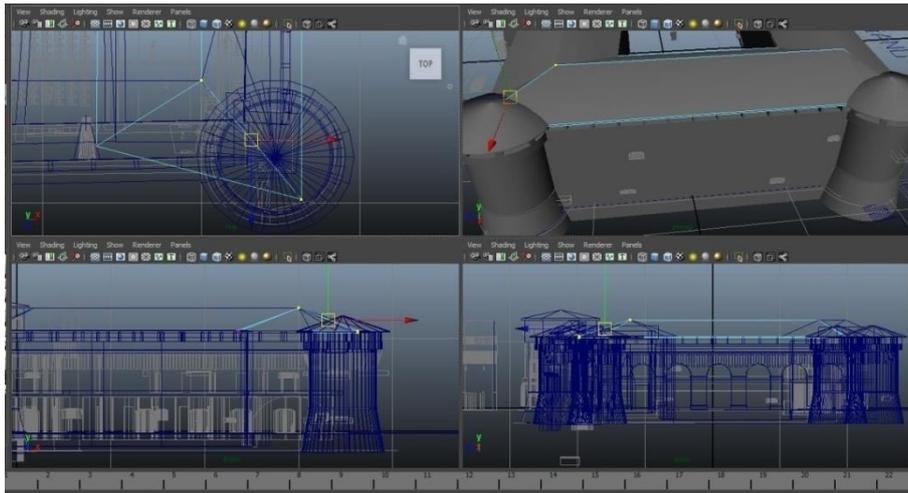
Per i camminamenti sulle varie cortine è stato invece necessario aggiungere dei nuovi poligoni, lavorati a loro volta per mezzo di estrusioni.



**Figura 27: Estrusione del secondo piano di una torre**

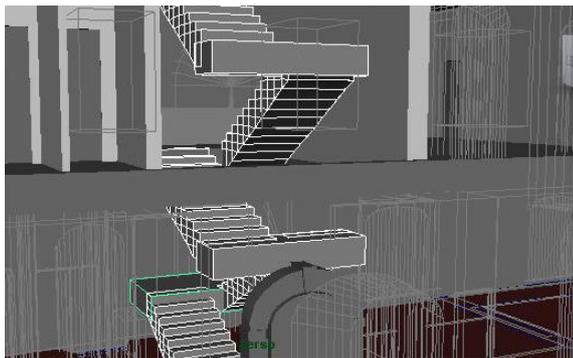
A questo punto è risultata evidente la necessità di lavorare in maniera differente la torre a base quadrata che sovrasta il ponte levatoio. Sia la sua facciata che gli interni sono infatti molto elaborati e la sua creazione in un unico poligono dà origine ad una forma estremamente articolata e pesante da gestire. Per questo la si è suddivisa utilizzando un poligono per ogni facciata e modellando il secondo piano a parte.

Il tetto è stato ottenuto con un misto di estrusioni e di spostamento dei vertici per adattarlo meglio alla forma irregolare della costruzione. Separatamente sono stati creati i tetti delle torri.



**Figura 28: Creazione del tetto.**

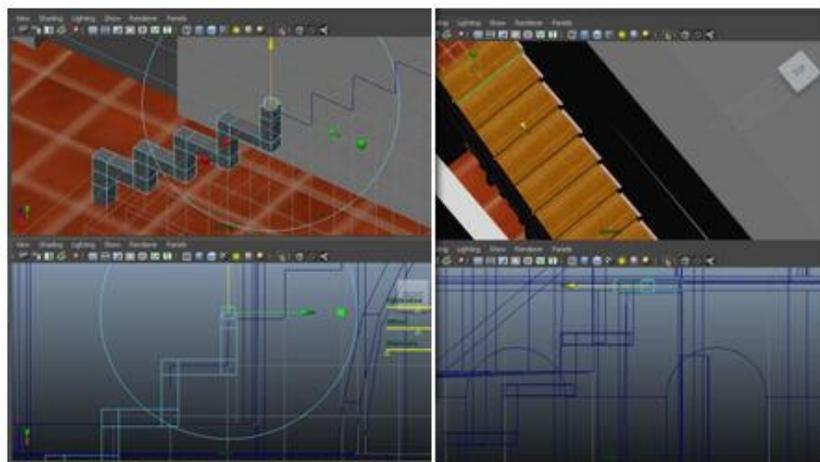
La rocca contiene numerose scale che sono state realizzate in due modi diversi a



**Figura 29: Scale primo tipo**

seconda delle specifiche. Per alcune è bastato creare un poligono con un numero di *subdivision* corrispondenti a quelle dei gradini della rampa e poi sollevarli man mano con delle estrusioni successive (Figura 29) e ritoccando le posizioni dei vertici delle facce inferiori. Per altre è stato necessario creare prima le guide laterali dei gradini con delle estrusioni, per poi posizionare manualmente i gradini stessi

ottenuti da un poligono duplicato più volte (Figura 30).



**Figura 30: Scale secondo tipo, guida laterale (a sinistra) e gradini (a destra).**

Una volta completata la struttura si è tornati alle finestre. Siccome queste sono quasi sempre diverse tra loro per forma e dimensioni, per ognuna di esse sono state realizzate separatamente le finestre vere e proprie, tramite la creazione di infissi e vetri. Ciò è stato fatto manipolando i poligoni da sottrarre per ottenere le aperture, utilizzando operazioni di scalatura, traslazione e sottrazioni booleane a seconda dei vari casi. Poi si è proceduto a realizzare le operazioni booleane lasciate in sospeso per creare le aperture nelle pareti.

Le porte volte all'esterno sono state realizzate in modo analogo, con l'accortezza di scomporle in più poligoni corrispondenti alla lunetta e alle varie ante in modo da consentire di aprirle nelle fasi successive.

In seguito si è passati ai dintorni. Strade e piazze sono state realizzate con dei piani ricchi di *subdivision*, eliminando le facce che interferivano con il fossato e gli interni della rocca. Il fossato ha richiesto una lavorazione più accurata utilizzando una superficie *nurbs* la cui forma è il risultato dell'utilizzo dello strumento *SculptGeometry*.

Sono poi stati aggiunti i dettagli, come lo stemma sopra la porta principale del Comune, dato da una sfera schiacciata o le doppie porte in vetro all'interno di alcune sale. Le bandiere sono formate da un cilindro per il palo e un piano con molte *subdivision* per la tela, dettaglio importante per l'animazione. I muretti che costeggiano il fossato sono stati realizzati estrudendo la faccia di un poligono seguendo un percorso disegnato da una linea *nurbs*.

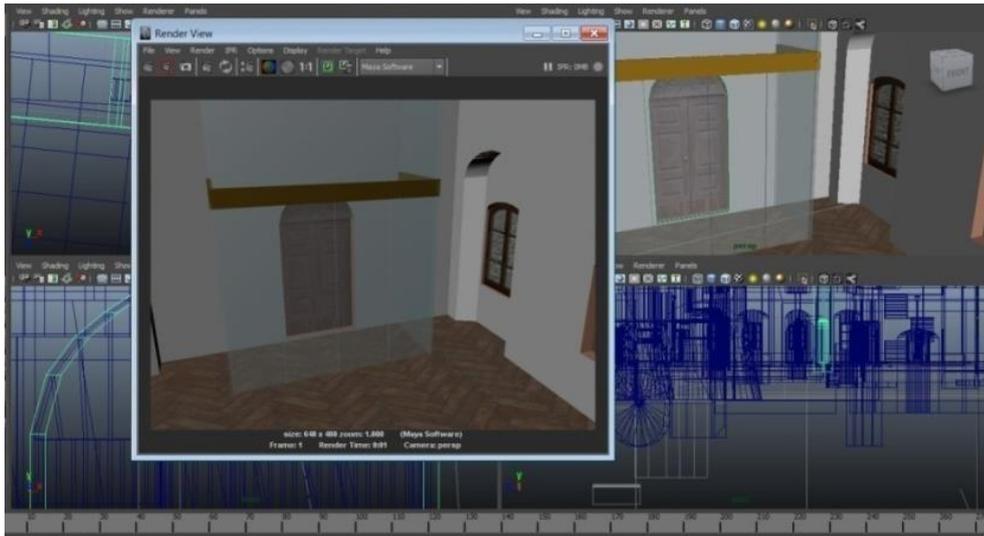
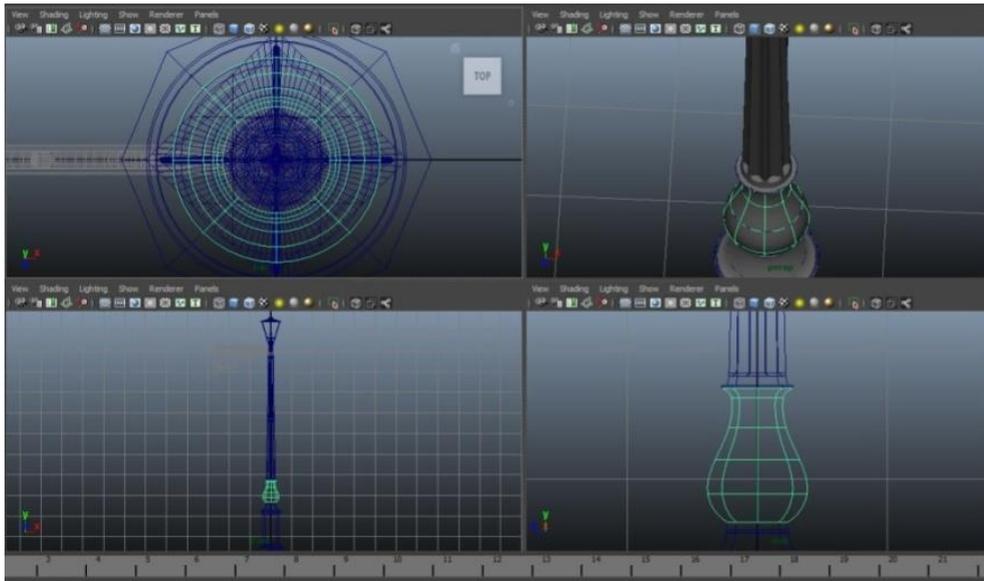


Figura 31: Porta interna in vetro

La stessa tecnica di estrusione è stata sfruttata per alcuni dei lampioni, specificamente quelli affissi ai muri, nella creazione dei riccioli decorativi che li caratterizzano. Per il resto i lampioni sono composti da delle superfici *nurbs* create principalmente usando la funzione *Loft* per collegare in ordine dei cerchi *nurbs* concentrici (o quadrati nella parte delle lanterne) di varie dimensioni e traslati a varie altezze e talvolta modificati spostandone i punti di controllo per creare scanalature e decorazioni nei pali.



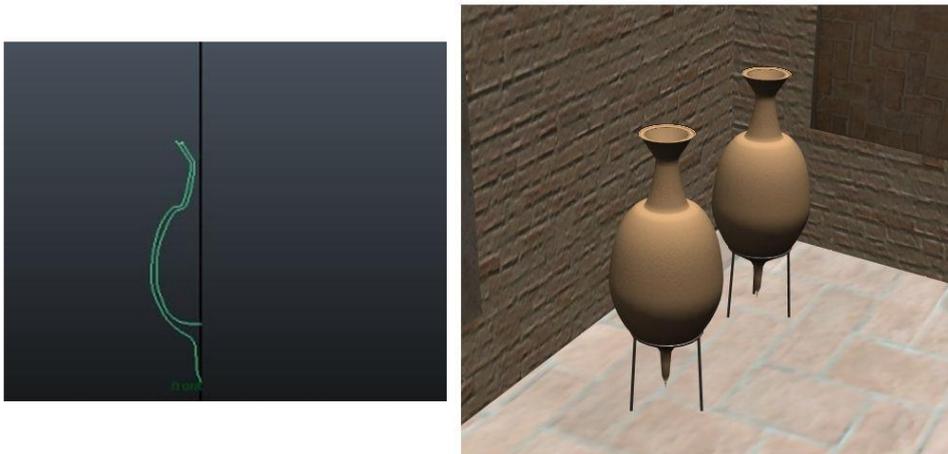
Figura 32: Decorazione di un lampione creata per mezzo di un'estrusione seguendo una linea nurbs



**Figura 33: Lampione formato da superfici nurbs create con lo strumento *Loft***

Terminata la struttura si è proceduto a realizzare gli interni.

All'interno del Museo Archeologico Tobia Aldini si è voluto ricreare l'ambiente riproducendo a grandi linee alcuni degli oggetti in esposizione, tra cui delle vetrinette ed il loro contenuto. A monete, bracciali e punte di freccia dalla forma essenziale si affiancano bottigliette, ciotole ed anfore create tracciandone il profilo con una linea *nurbs* e sfruttando la funzione *Revolve* per ottenerne la superficie.

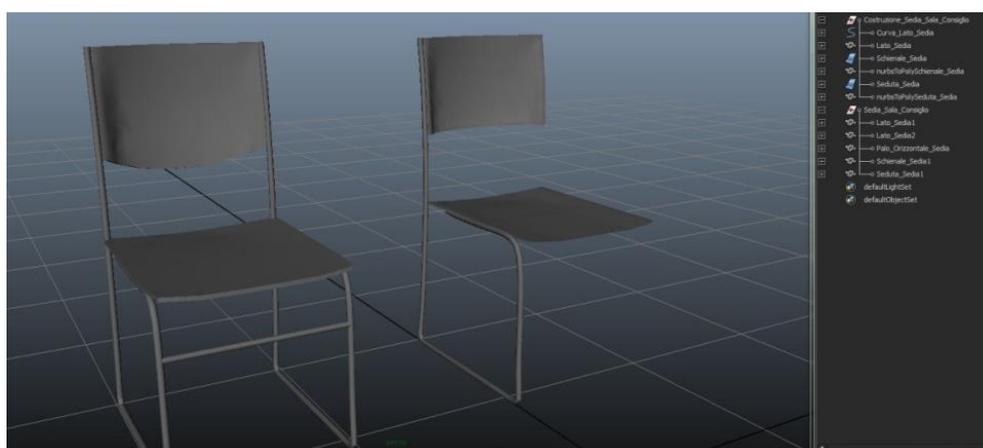


**Figura 34: Profilo nurbs di un'anfora e rendering del risultato finale.**



**Figura 35: Profili nurbs di una ciotola e di alcune bottigliette e rendering del risultato finale.**

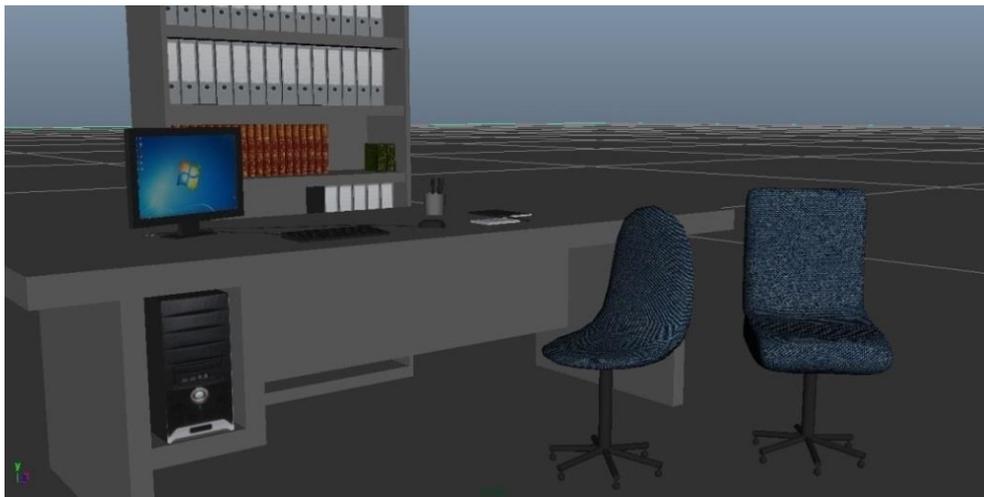
L'arredamento della Sala del Consiglio è stato riprodotto usando un misto di tecniche. I tavoli sono realizzati con dei poligoni e lo strumento *polyExtrude*, mentre per le sedie si sono sfruttate le *nurbs*. In particolare, la seduta e lo schienale sono superfici *nurbs* a cui si è data la forma leggermente concava con lo strumento *SculptGeometry*, prima di convertirle in poligoni per gli ultimi ritocchi. I restanti elementi della sedia sono il frutto dell'estrusione di un poligono lungo un percorso disegnato con una curva *nurbs*.



**Figura 36: Sedie della Sala del Consiglio. A destra quella incompleta ancora formata da superfici *nurbs*, a sinistra il risultato finale.**

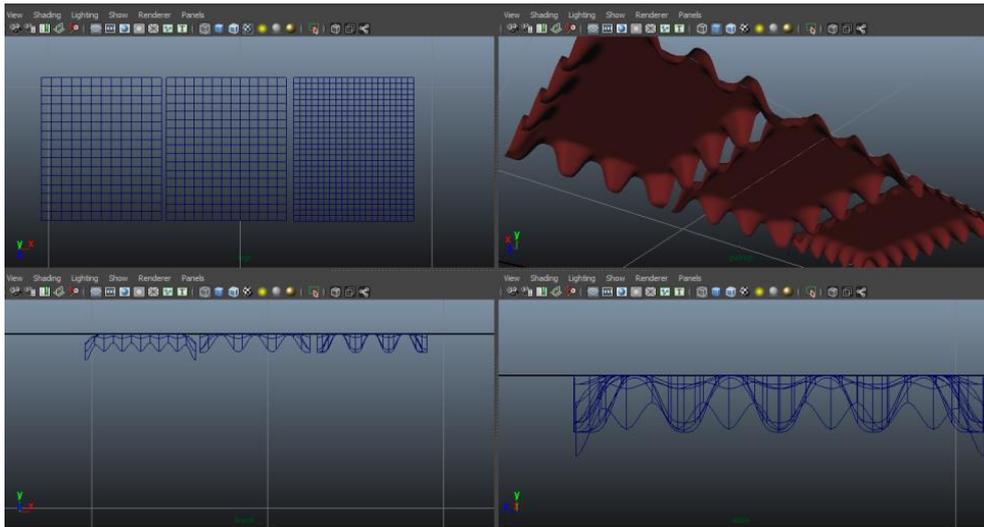
Per la Sala Mostre è stata realizzata una semplice esposizione sfruttando le immagini della storia della rocca presenti in questa tesi, mentre i mobili per gli uffici

del comune sono stati anch'essi creati con un misto di tecniche e poi duplicati con piccoli accorgimenti per adattarli al locale in cui sono stati collocati. Interessanti sono state nuovamente le sedie, stavolta imbottite e su ruote. La seduta è realizzata a partire da un cubo, modificato con scalature ed estrusioni in modo che riproduca grossolanamente l'aspetto della sedia e poi 'arrotondato' con lo strumento *smooth*. Il diverso numero di *subdivision* specificate per il cubo iniziale determina l'aspetto più tondeggiante o squadrato del risultato.



**Figura 37: Alcuni mobili realizzati per gli uffici.**

Infine, sono state realizzate le decorazioni rinascimentali ancora presenti nel soffitto di una delle stanze al primo piano. Si è scelto di creare una sorta di controsoffitto tramite un piano *nurbs*, selezionando i vertici di controllo del perimetro secondo uno schema e traslandoli verso il basso. Le proprietà delle *nurbs* si occupano del resto.



**Figura 38: Prove per la realizzazione del "controsoffitto". Quello centrale è poi stato importato nel file contenente la ricostruzione della rocca ed adattato alla sala.**

### 4.3 Texturing

Per praticità i primi materiali sono stati inseriti già durante la fase di modellazione per rendere più facile distinguere gli oggetti durante la stessa.

Per le pareti, è stato sufficiente assegnare un *Lambert* specificandone il colore, mentre il granito è un effetto offerto dal programma che consente anche di adattarlo alle proprie specifiche esigenze, permettendo di ricreare materiali diversi come la pietra di alcuni gradini o l'asfalto della strada. Il vetro e i metalli sono invece ricavati variando le proprietà di un *Blinn*.

Per quanto riguarda le texture vere e proprie, il lavoro si è basato principalmente sulle foto scattate sul luogo.

Dove è stato possibile si è sfruttato Internet per reperire delle textureseamless (cioè coi bordi opposti che coincidono quando vengono affiancati) che potessero essere ritoccate sfruttando Paint Shop Pro e Photoshop per renderle il più simili possibile alle originali



**Figura 39:** A sinistra, foto del pavimento della Sala Mostre. Al centro, una texture seamless scaricata da Internet. A destra, la texture ritoccata con Photoshop per modificarne i colori e con Paint Shop Pro per aggiungere l'effetto consunto dei bordi.

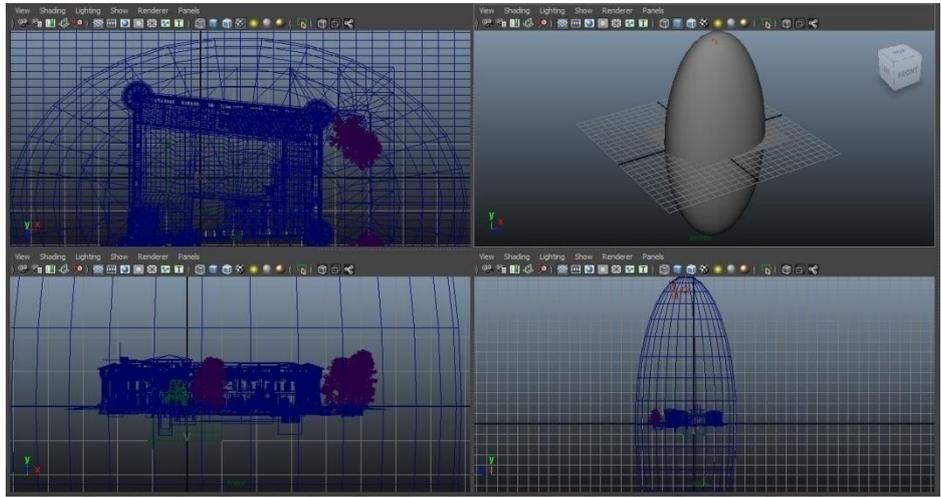


**Figura 40:** Texture di una porta interna della Rocca ricavata da una fotografia.

In altri casi, le texture sono state create manipolando le foto stesse, tagliandone alcune sezioni e ritoccandole leggermente, come ad esempio quella della porta interna nella figura 40. Ricavata da una fotografia: se ne è corretta la prospettiva e se ne è creata una copia riflessa per il lato opposto, eliminando la targhetta affissa in alto sul lato esterno della stanza.

Tra queste, la texture che ha richiesto la maggiore elaborazione è stata senza dubbio quella per l'envelopingsphere che riproduce i dintorni della rocca. Si tratta di una serie di fotografie unite in parte sfruttando una funzione intrinseca del programma ZoomBrowser EX e in parte manualmente in Paint Shop Pro e Photoshop, con l'aggiunta di una texture adeguata per riprodurre il cielo. Ci si è accertati che l'azzurro non comparisse a livello del terreno "ampliando" la parte inferiore delle foto con lo strumento *push* in Paint Shop Pro ed il suo equivalente *fluidifica* in Photoshop. Man mano che venivano aggiunte le componenti è stato necessario ridurre l'immagine risultante di dimensioni fino ad ottenere una texture che Maya potesse gestire, seppure sempre molto pesante per non perdere troppa qualità. È stata una degli ultimi elementi ad essere inseriti proprio per questo motivo. Inoltre, per evitare che gli edifici che non compaiono esattamente

direttamente di fronte alla telecamera apparissero inclinati nel rendering si è scalata la sfera, allungandola in corrispondenza dell'asse y.



**Figura 41: Envelopingsphere**



**Figura 42: Texture dell'envelopingsphere**

#### 4.4 Semplificazione e pulizia del modello

In preparazione al trasferimento in Unity 3D, sono stati eliminati dal modello tutti gli elementi che non influivano sul risultato visivo finale. In primo luogo si è provveduto a rimuovere le fasi di lavorazione lasciate dalle operazioni booleane, lasciandone solo i risultati finali. Analogamente, si sono cancellati gli elementi di costruzione lasciati da parte e congelati in modo che non disturbassero il resto del lavoro.

Dato che Unity non supporta le NURBS è stato necessario convertire tutte le superfici e le curve di questo tipo in poligoni, mentre gli alberi realizzati con painteffects sono scomparsi, richiedendo una fase di lavorazione in più all'interno del nuovo programma.

Sempre per una specifica di Unity per cui un oggetto è visibile solo nella direzione indicata dalle normali, è stato necessario invertire le normali delle facce dell'envelopingsphere di modo che fosse visibile dalla telecamera che si muove al suo interno, mentre risulta invisibile dall'esterno.

Il file è stato poi già esportato in formato .fbx per non rendere troppo gravoso il processo di importazione in Unity.



**Figura 43: Rendering della torre del ponte levatoio.**



Figura 44: Redering del ponte.



Figura 45: Rendering di uno scorcio di Piazza Fratti.



**Figura 46: Rendering di un interno del museo.**



**Figura 47: Rendering della cortina occidentale.**

## CAPITOLO 5

### REALIZZAZIONE DELLA VISITA VIRTUALE

#### 5.1 Passaggio a Unity

Il modello elaborato in Maya e predisposto al passaggio è stato importato come un Asset in un nuovo progetto di Unity, aggiustando il percorso per il reperimento di quelle texture che, seppur ancora presenti ed applicate alle primitive, non venivano più visualizzate sul modello.

Si è quindi aggiustata la scala alle unità di misura di Unity, operazione che risulterà utile in seguito quando verrà inserito il *CharacterCollider*.

Una volta affrontati questi problemi immediati, si è passati ad una tecnica alternativa per realizzare gli alberi sfruttando il *Tree Editor* di Unity. Questo permette, partendo dal tronco (qui considerato un ramo singolo), di aggiungere ricorsivamente rami e in seguito, foglie. Attraverso la modifica dei parametri dati, come ad esempio il *seed* per la generazione dei rami o la loro inclinazione, risulta semplice diversificare gli alberi l'uno dall'altro e le texture per la corteccia e le foglie aiutano a distinguere il tipo di albero.

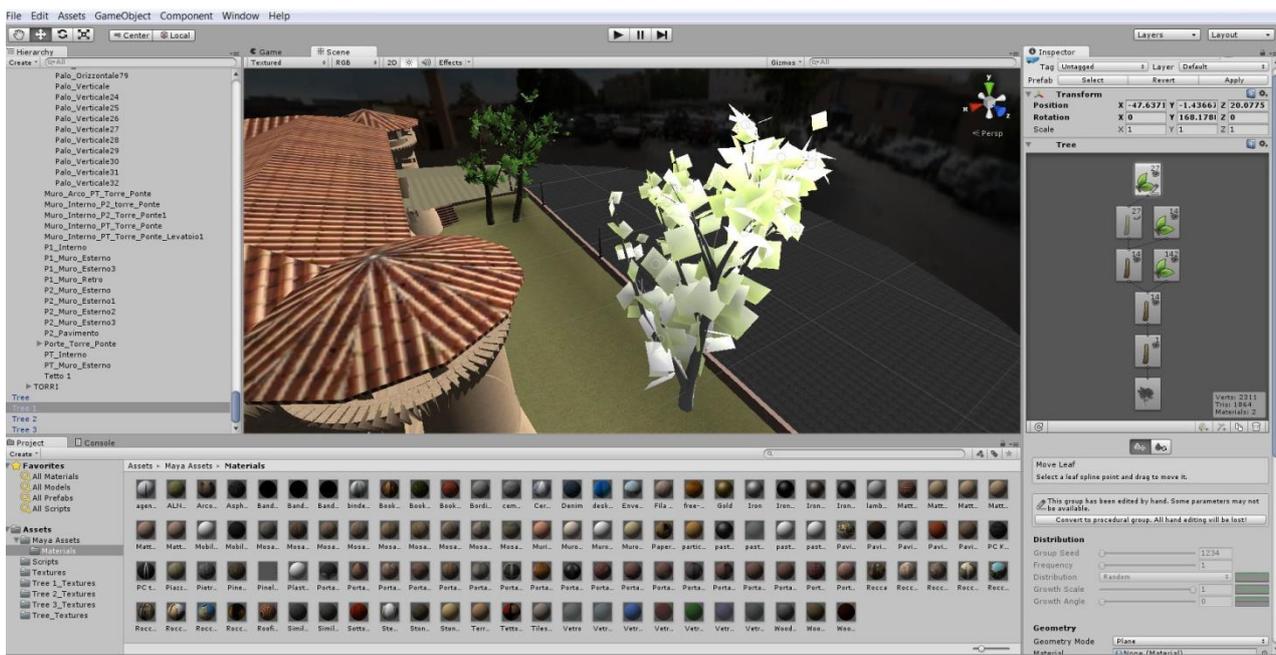


Figura 48: Tree Editor. In primo piano un albero in fase di lavorazione; più distanti due già ultimati.

L'ultimo elemento che si è rivelato necessario rifare a causa del passaggio sono state le tre bandiere che sventolano su uno dei camminamenti. In Maya ciascuna di esse era stata creata con un'animazione a partire da una primitiva di *nCloth*, ma questa ha perso tutte le sue proprietà nel trasferimento. È stata quindi eliminata e sostituita con un nuovo *GameObject* di tipo *Cloth*, le cui proprietà sono state aggiustate per riprodurre l'effetto del vento. Da qui si è duplicato l'oggetto e si è proceduto a differenziare le tre bandiere tramite le texture ed ancorando i piani rappresentanti i tessuti alle rispettive aste sfruttando l'apposita funzione di Unity.



Figura 49: Le bandiere realizzate in Unity3D.

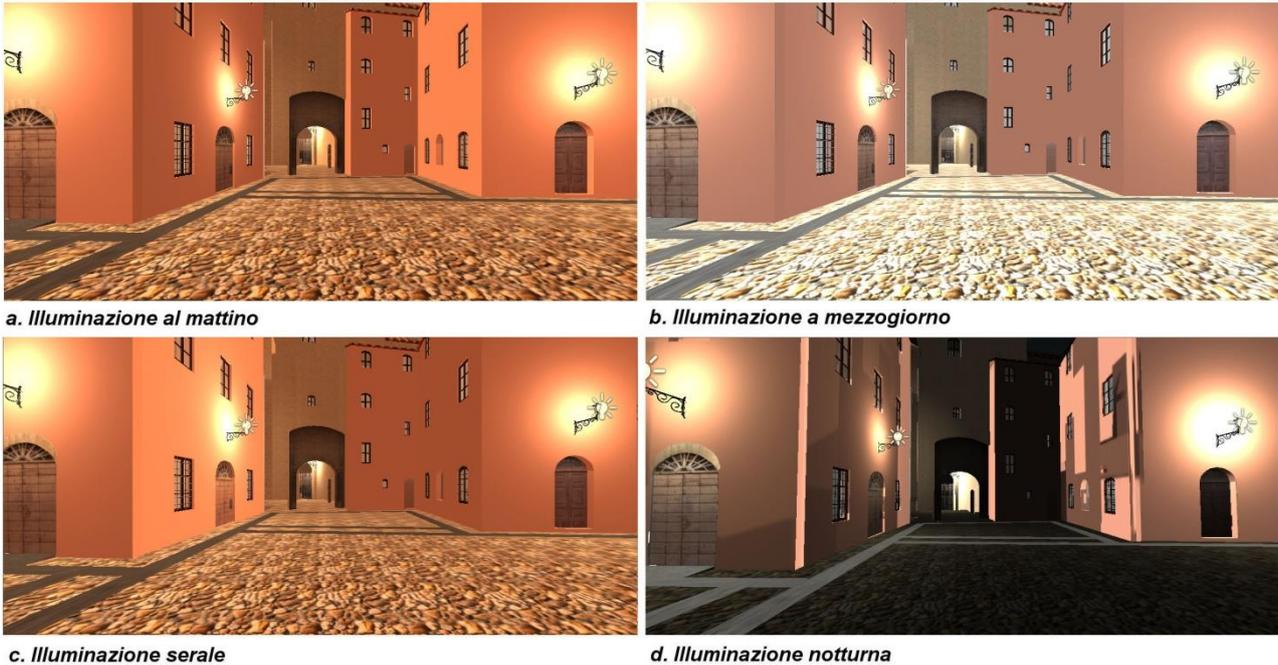
## 5.2 Lighting

La scena contiene già un'*ambient light* di default, impostata ad una bassa intensità. È stata lasciata per assicurarsi che gli interni avessero un minimo di illuminazione e ad essa è stata aggiunta una *directional light* che simuli la luce solare.

Entrambe sono state poi modificate all'interno di uno script per riprodurre l'illuminazione dei vari momenti della giornata, cambiando dall'uno all'altro alla pressione del tasto "g" della tastiera. In primo luogo viene modificato il colore, lo stesso per entrambe le luci ad eccezione dell'illuminazione notturna. In quest'ultima la *directional light* è nera, il che equivale a non presente, mentre l'*ambient*

*light* viene impostata a un grigio molto scuro, altrimenti nessuna parte del modello risulterebbe visibile, riducendo la scena ad un'immagine completamente nera.

Oltre al colore si modifica l'inclinazione della *directional light*, orientandola in modo che riproduca la direzione del sole e per incrementare la somiglianza con l'astro se ne aumenta leggermente l'intensità quando il momento della giornata corrisponde a mezzogiorno.



**Figura 50: L'illuminazione nelle varie fasi della giornata.**

Per l'illuminazione notturna si sono aggiunte delle *pointlights* in corrispondenza dei lampioni, il che comporta un elevato quantitativo di calcoli impossibile da realizzare in tempo reale. Unity ricorre ad uno stratagemma e le calcola come *vertexlights* invece che come *pixel lights* (calcola l'illuminazione ai vertici di ogni mesh e ricava i punti intermedi della superficie per interpolazione invece di ripetere il calcolo per ogni pixel), ma il risultato non è altrettanto apprezzabile.

Dato che la struttura è statica (con la sola eccezione di bandiere e porte) è stato possibile ovviare al problema creando una *lightmap*. Impostando tutti gli elementi fissi della scena, incluse le *pointlights*, come *static*, si è poi potuto precalcolare queste luci tramite l'istruzione *Bake scene*, e poi applicare la *lightmap* alle mesh quasi come se fosse una texture.

```

1 using UnityEngine;
2 using System.Collections;
3
4 public class LightScript : MonoBehaviour
5 {
6     Color32[] colori = new Color32[4];
7     bool buttonG = false;
8     int count = 0;
9     public Light lt;
10    Vector3[] rotation = new Vector3[4];
11
12    Color32 alNight = new Color32(); // colore notturno ambient light
13
14    // Use this for initialization
15    void Start ()
16    {
17        //sunrise
18        colori [0].r = 182;
19        colori [0].g = 126;
20        colori [0].b = 91;
21        //noon
22        colori [1].r = 192;
23        colori [1].g = 191;
24        colori [1].b = 193;
25        //sunset
26        colori [2].r = 182;
27        colori [2].g = 126;
28        colori [2].b = 91;
29        //night
30        colori [3].r = 0;
31        colori [3].g = 0;
32        colori [3].b = 0;
33
34        //sunrise
35        rotation [0].x = 260;
36        rotation [0].y = 0;
37        rotation [0].z = 0;
38        //noon
39        rotation [1].x = 50;
40        rotation [1].y = 0;
41        rotation [1].z = 0;
42        //sunset
43        rotation [2].x = 50;
44        rotation [2].y = 0;
45        rotation [2].z = 0;
46        //night - irrilevante perchè la luce è nera
47        rotation [3].x = 0;
48        rotation [3].y = 0;
49        rotation [3].z = 0;
50

```

```

51     alNight.r = 20;
52     alNight.g = 20;
53     alNight.b = 20;
54
55     lt = GetComponent<Light>();
56     lt.color = colori[0];
57     lt.intensity = 0.5f;
58
59     RenderSettings.ambientLight = colori[0];
60 }
61
62 // Update is called once per frame
63 void Update ()
64 {
65     buttonG = Input.GetKeyDown (KeyCode.G);
66     if(buttonG == true)
67     {
68         count++;
69         count = count % 4;
70         lt.color = colori[count];
71         transform.Rotate(rotation[count]);
72
73         if(count == 1)
74             lt.intensity = 0.6f;
75         else
76             lt.intensity = 0.5f;
77
78         //ambient light
79         RenderSettings.ambientLight = colori[count];
80
81         if(count == 3)
82             RenderSettings.ambientLight = alNight;
83
84     }
85 }
86 }
87 }
88

```

Figura 51: Script dedicato alle operazioni di Lighting.

### 5.3 Fisica e apertura porte

Il passo successivo è stato preparare il modello per consentire il movimento al suo interno in base alle leggi della fisica, come ad esempio rendere impossibile il passaggio attraverso i muri.

Principalmente questo si è tradotto nel dotare numerosi elementi della scena *colliders*, componenti che ne definiscono la forma allo scopo di calcolare le collisioni, come suggerito dal nome stesso. Tra i vari tipi di *colliders* messi a disposizione da Unity ne sono stati utilizzati tre, scelti a seconda del caso. Dove possibile si è preferito utilizzare dei *box colliders* o dei *capsule colliders* per rendere l'elaborazione meno costosa, in altri casi si è utilizzato un *mesh collider*, come nel caso di un muro contenente una porta per consentire il passaggio attraverso la stessa. Per evitare sprechi di tempo e di lavoro per la CPU, tutti gli elementi in cui il "visitatore" non entrerà in contatto, come ad esempio il tetto o le decorazioni del soffitto, non sono stati dotati di alcun collider.

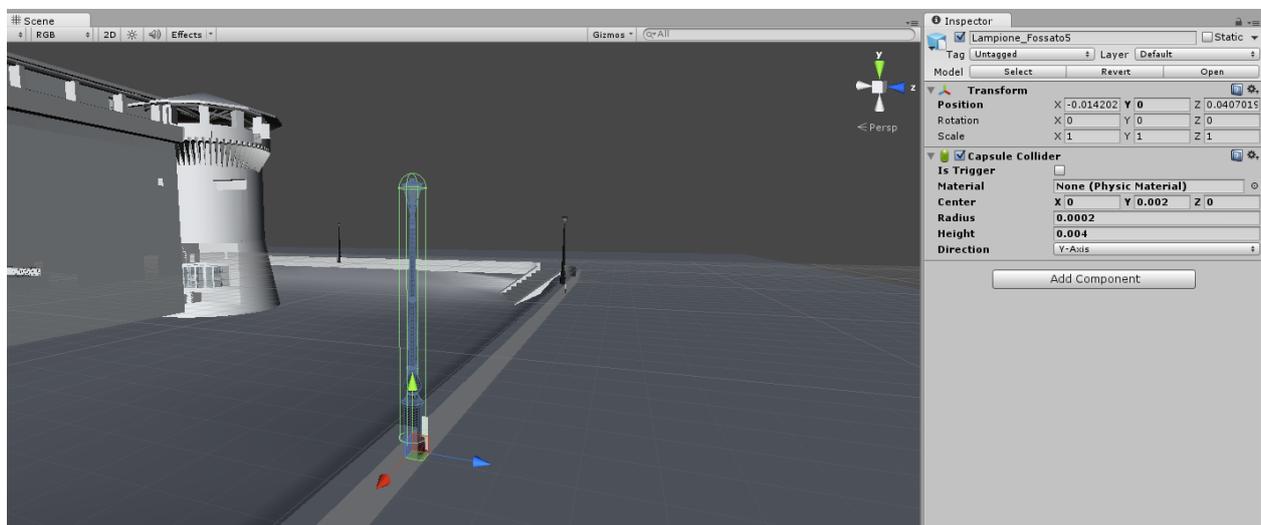


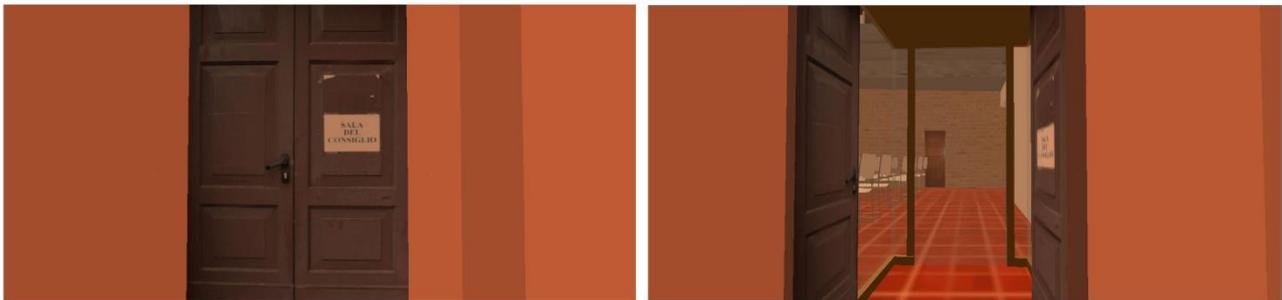
Figura 52: Inserimento di un *capsule collider* (in verde) attorno ad un lampione.



Figura 53: Uso di svariati *box collider* (in verde) intorno agli arredi di un ufficio.

Per permettere l'apertura delle porte è stato necessario un lavoro più articolato. Il muro circostante è sempre dotato di un *mesh collider* per consentire al visitatore di attraversare il vano della porta, mentre quest'ultima è stata dotata di uno script che la apra ogni volta che il visitatore si avvicina.

La sua presenza viene riconosciuta grazie ad un *collider* che include l'area circostante alla porta ed impostato come *trigger*. Questo significa che invece di essere riconosciuto come impossibile da attraversare, la collisione lancia un segnale che fa attivare lo script, aprendo la porta e in seguito richiudendola quando il visitatore lascia l'area.



**Figura 54: la porta si apre all'avvicinarsi del visitatore.**

Dato che Unity imposta automaticamente i pivot al centro dell'oggetto, per rendere possibile la rotazione della porta intorno ai cardini è stato necessario renderla oggetto figlio di un oggetto vuoto con il pivot nella posizione giusta. Così facendo basta ruotare l'oggetto genitore e la porta si muove di conseguenza.

A seconda del senso di rotazione della porta (orario o antiorario) sono stati necessari due diversi script, identici in tutto e per tutto eccetto che per il segno che determina la rotazione.

```

1 using UnityEngine;
2 using System.Collections;
3
4 public class DoorScript1 : MonoBehaviour {
5
6     float rotation;
7
8     // Use this for initialization
9     void Start ()
10    {
11        rotation = 0;
12    }
13
14    void OnTriggerEnter(Collider other)
15    {
16        rotation = 90f;
17        transform.Rotate(0.0f, rotation, 0.0f);
18    }
19
20
21    void OnTriggerExit(Collider other)
22    {
23        rotation = -90f;
24        transform.Rotate(0.0f, rotation, 0.0f);
25    }
26 }
27

```

Figura 55: Uno dei due script per l'apertura delle porte.

La stessa tecnica e gli stessi script sono stati utilizzati anche per rendere apribili gli armadietti all'interno degli uffici.



Figura 56: Apertura armadietto all'avvicinarsi del visitatore. In verde è visibile il *collider* che funge da *trigger*.

## 5.4 Movimento

Paragonando questa visita virtuale ad un videogame l'effetto che si vuole ottenere è quello di un gioco in prima persona, in cui il personaggio (in questo caso il visitatore) si muove all'interno dell'ambiente seguendo i comandi del giocatore. Il personaggio stesso non è visibile, permettendo quindi di realizzarlo tramite un *EmptyGameObject*.

Per comodità lo si è posizionato nell'origine, mentre il *Character Collider* ad esso associato è stato traslato lungo l'asse y di 0.76. Questo perché il *GameObject* di base si trova al centro del *collider* in questione, che è alto 1.5 unità. Traslando quest'ultimo verso l'alto di una unità lo si porta al livello del terreno e l'aggiuntivo 0.01 dà la certezza che si trovi sopra il terreno ad ogni momento, eliminando il rischio che il programma lo consideri dentro al terreno portando ad una caduta del personaggio attraverso il piano d'appoggio e nel vuoto al di sotto.

La telecamera è stata posizionata nell'origine e da qui si è adattata la sua posizione ad un'altezza rispetto al terreno che riproducesse il punto di vista di un visitatore per creare l'illusione di una visita in prima persona.

Ci sono vari modi per fare sì che la telecamera segua il personaggio nei suoi spostamenti. Si può utilizzare uno script affinché lo segua passo a passo o, più semplicemente, si può sfruttare l'ereditarietà e rendere la telecamera un *GameObject* figlio del *GameObject* personaggio. Così facendo lo seguirà automaticamente.

Al personaggio è stato in seguito aggiunto come componente uno script. Questo, non appena chiamato, nasconde il cursore del mouse, eliminando una fonte di distrazione durante la visita. Dopodiché procede a descrivere le reazioni del personaggio alla pressione di determinati tasti e ad alcuni movimenti del mouse.

---

```

1 using UnityEngine;
2 using System.Collections;
3
4 public class FirstPersonPlayer : MonoBehaviour {
5
6     public float movespeed = 5.0f;
7     public float rotationspeed = 10.0f;
8
9     void Start ()
10    {
11        Screen.lockCursor = true;
12        //nasconde il cursore del mouse. Ricompare premendo Esc.
13    }
14
15    void Update ()
16    {
17        //rotation (yaw)
18
19        float rotation = Input.GetAxis ("Mouse X");
20        rotation = rotation * rotationspeed;
21
22        transform.Rotate(0.0f, rotation, 0.0f);
23
24        // movement
25
26        float moveHorizontal = Input.GetAxisRaw ("Horizontal");
27        float moveVertical = Input.GetAxisRaw ("Vertical");
28
29        Vector3 movement = new Vector3 (moveHorizontal, 0.0f, moveVertical);
30        movement = transform.rotation * movement;
31        //fa in modo che il movimento sia basato sull'asse locale invece
32        // che su quello globale
33
34        CharacterController cc = GetComponent<CharacterController> ();
35
36        cc.SimpleMove (movement * movespeed);
37    }
38 }

```

**Figura 57: Script che gestisce il *GameObject* rappresentante il visitatore, in particolare gli spostamenti e la rotazione di tipo *yaw*.**

### 5.4.1 Spostamento

Tramite lo script assegnato al personaggio gli si è data la possibilità di muoversi catturando l'input trasmesso attraverso i tasti WASD oppure con le frecce direzionali. Le funzioni `Input.GetAxis ("Horizontal")` e `Input.GetAxis ("Vertical")` restituiscono dei valori positivi o negativi che segnalano la presenza di movimento lungo quel determinato asse. Il valore 0 segnala l'assenza di input e di

conseguenza l'assenza di movimento, altrimenti il segno specifica in che direzione ci si sta muovendo lungo quell'asse.

Le varianti `Input.GetAxisRaw ("Horizontal")` e `Input.GetAxisRaw ("Vertical")` qui preferite accettano solo valori interi, permettendo un controllo leggermente più preciso del movimento.

I valori ottenuti sono stati poi memorizzati all'interno di un vettore(x, y, z) in corrispondenza degli elementi x e z. Moltiplicando questo vettore per la rotazione dell'oggetto, inoltre, si fa in modo che i valori si riferiscano all'asse locale invece che all'asse globale della scena.

Il vettore ottenuto viene applicato al *Character Controller* dopo essere stato moltiplicato per la velocità di movimento, un valore determinato durante la fase di test dello script per rendere l'effetto più gradevole.

Il metodo `SimpleMove` è particolarmente adatto per realizzare i movimenti perché regolarizza la velocità on-screen indipendentemente dal numero di frame al secondo e non si incorre nei suoi limiti, non essendo previsti movimenti in verticale, come ad esempio salti, che non sarebbero possibili con questo metodo.

#### 5.4.2 Rotazione (yaw)

Dato che la rotazione viene chiamata in causa per determinare l'asse locale durante il movimento, essa deve essere determinata prima di quest'ultimo all'interno sempre dello script del personaggio. Il comando viene trasmesso attraverso il movimento del mouse e moltiplicato per una velocità di rotazione, determinata allo stesso modo di quella di movimento.

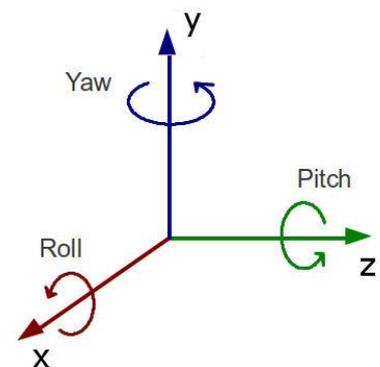


Figura 58: Tipi di rotazione

Il valore ottenuto viene quindi applicato direttamente alla rotazione dell'oggetto rendendo superfluo l'utilizzo del *Character Controller*. Notare che la rotazione avviene solo intorno all'asse y, cioè quello verticale, consentendo quindi al *GameObject* di ruotare su se stesso e cambiare direzione.

### 5.4.3 Rotazione (roll)

Per realizzare questo tipo di rotazione occorre tenere conto di una particolarità del *Character Controller*, cioè del fatto che questo ha sempre una posizione verticale. Qualsiasi rotazione intorno all'asse x (*roll*) deve quindi essere applicata alla telecamera. Lo script contenente il codice è quindi un componente di quest'ultima.

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class FirstPersonCamera : MonoBehaviour {
5
6     public float rollspeed = 5.0f;
7     public float range = 50.0f; //gradi di rotazione consentiti
8     float roll = 0;
9
10    void Start ()
11    {
12
13    }
14
15    void Update ()
16    {
17
18        roll -= Input.GetAxis ("Mouse Y") * rollspeed;
19        roll = Mathf.Clamp (roll, -range, range);
20        //limita la rotazione (roll) all'interno del range
21
22        transform.localRotation = Quaternion.Euler (roll, 0.0f, 0.0f);
23    }
24 }
```

**Figura 59: Script assegnato alla telecamera, che gestisce la rotazione di tipo *roll*.**

La tecnica è la stessa utilizzata per determinare la rotazione intorno all'asse y (*yaw*) con la differenza che la rotazione è applicata ad un asse diverso. Tramite il metodo *Clamp* viene inoltre introdotto un range di rotazione, che pone un valore massimo e uno minimo a quanto la visuale può essere spostata verso l'alto o il basso.



Figura 60: Massima rotazione (*roll*) verso l'alto.

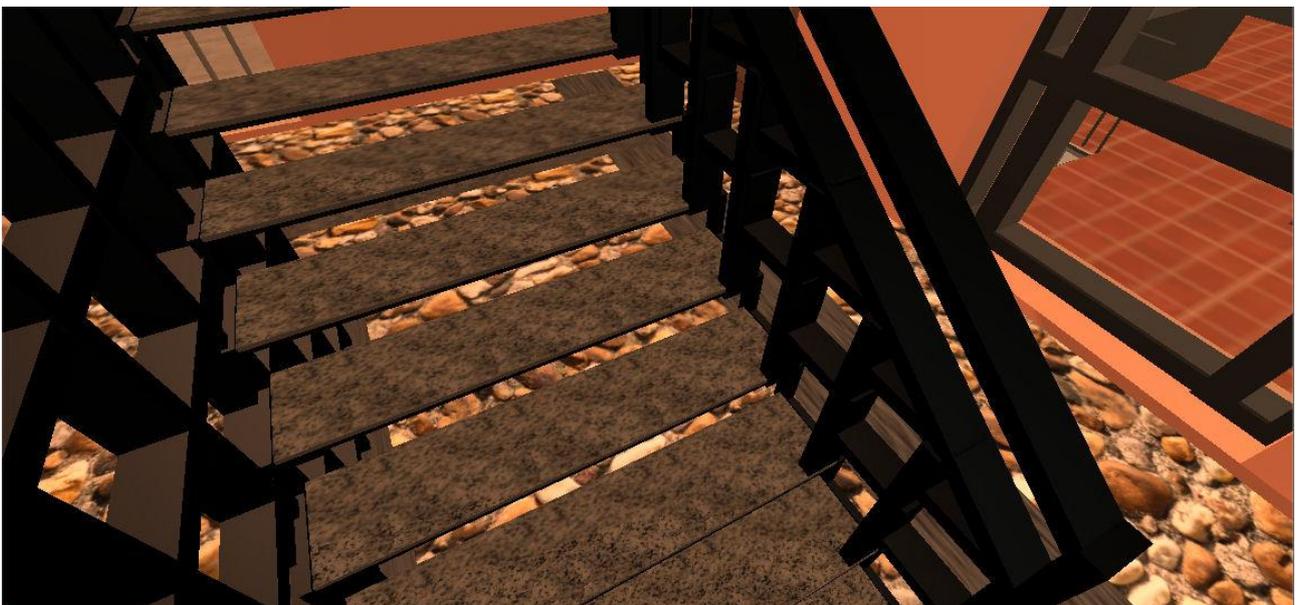


Figura 61: Massima rotazione (*roll*) verso il basso.

## 5.5 Creazione dell'eseguibile

Una volta completato il lavoro in Unity3D e dopo aver ampiamente testato la visita virtuale all'interno dell'*editor*, si è passati alla creazione dell'eseguibile, che rende possibile utilizzare il programma finale anche su una macchina su cui non è installato Unity.

Non essendoci necessità specifiche per la pubblicazione ci si è limitati a creare un eseguibile per Windows, anche se Unity offre una gamma di possibilità molto più vasta.

Tutti gli elementi sono stati inseriti all'interno di una singola scena, semplificando ulteriormente il lavoro di *Build*. In effetti è bastato selezionare le opzioni finora elencate e lanciare la procedura, ottenendo un file .exe accompagnato da una cartella dati contenente tutte le risorse necessarie per il funzionamento dell'applicazione.

## BIBLIOGRAFIA E SITOGRAFIA

### LIBRI

- [Ald99] Aldini T., *Storia di Forlimpopoli*, Cesena, Società Editrice "Il Ponte Vecchio", 1999
- [Ber70] Berardi D. e altri, *Rocche e Castelli di Romagna*, vol. II, Bologna, Edizioni ALFA, 1970, pag 258-265
- [CG05] Colombini P., Grottanelli de' Santi E., *L'Italia. Emilia Romagna*, Milano, Touring Editore, 2005, pag 864-866
- [Col02] Colombo D. e altri, *Romagna*, Paderno Dugnano (MI), Fininternet, 2002, pag 134
- [SCMS05] Scateni R., Cignoni P., Montani C., Scopigno R., *Fondamenti di grafica tridimensionale interattiva*, Milano, McGraw Hill, 2005
- [Sol12] Solarsky Chris, *Drawing Basics and Video Game Art*, New York, Watson Guptill, 2012
- [Ted92] Tedeschini M., *Atlante Romagnolo. Dizionario alfabetico dei 76 comuni*, Bologna, Poligrafici Editoriale, 1992, pag 49-50

### SITI

- [Bar14] Bartoli S., "Rocca Hordelaffa", <http://www.comune.forlimpopoli.fc.it/servizi/menu/dinamica.aspx?idArea=17175&idCat=19029&ID=17182&TipoElemento=categoria>
- [Bro13] Brodtkin J., "How Unity 3D Became a Game Development-Beast", 2013, <http://news.dice.com/2013/06/03/how-unity3d-become-a-game-development-beast/>
- [Car03] Carlson W., "A Critical History of Computer Graphics and Animation", 2003, <http://design.osu.edu/carlson/history/lessons.html>
- [CNE11] Carrozzino M., Neri V., Evangelista C., Bergamasco M., "Piazze reali e piazze virtuali. Visualizzazione interattiva della ricostruzione virtuale di

- piazze famose come strumento di conoscenza", 2011,  
<http://www.mediageo.it/ojs/index.php/archeomatica/article/view/65>
- [DJL15] Dionisio G., Jasink A.M., Lazzi G., Licari D., "Restauro Archeologico e Restauro Librario: due diverse applicazioni del restauro virtuale per la conservazione del patrimonio dei beni culturali", 2015,  
<http://www.mediageo.it/ojs/index.php/archeomatica/article/view/895>
  - [Dre13] Dreher T., "History of Computer Art", 2013, [http://iasl.uni-muenchen.de/links/GCA\\_Indexe.html](http://iasl.uni-muenchen.de/links/GCA_Indexe.html)
  - [Her13] Heritagedaily, "New method for reconstructing historic buildings in 3D", 2013, <http://www.heritagedaily.com/2013/12/new-method-for-reconstructing-historic-buildings-in-3d/100533>
  - [Lev01] Levy R., "Computer Reconstruction: Temple Site at Pimai - Introduction", 2001, <http://www.phimai.ca/intro.htm>
  - [Lor12] Lorenzini M., "Semantic approach to 3D Historical Reconstruction", 2012, <http://www.isprs.org/proceedings/XXXVIII/5-W1/pdf/lorenzini.pdf>
  - [Min13] Minelli M.R., "Realtà aumentata, ricostruzioni 3D, e 'teche parlanti' svelano i tesori etruschi di Cerveteri", 2013,  
<http://www.mediageo.it/ojs/index.php/archeomatica/article/view/335>
  - [Mor14.1] Mori L. "Godzilla Week! MPC reveals their tips and tricks to creating a monster", 2014,  
<http://www.3dartistonline.com/news/2014/06/godzilla-week-mpc-reveals-their-tips-and-tricks-to-creating-a-monster/>
  - [Mor14.2] Mori L. "Godzilla Week part 2: Rigging, Animation and more", 2014, <http://www.3dartistonline.com/news/2014/07/godzilla-week-part-2-rigging-animation-and-more/>
  - [OA09] Olivi R., Agostini A., "Riqualificazione di Piazza Garibaldi, con parziale rifacimento dei sottoservizi", 2009,  
<http://europaconcorsi.com/albo/78-Ordine-degli-Architetti-Pianificatori-Paesaggisti-e-Conservatori-della-provincia-di-Forl-Cesena/projects/111759-Piazza-Garibaldi/print>

- [PK10] Papadopoulus K., Kefalaki E., "At the Computer's Edge. The Value of Virtual Constructions to the Interpretation of Cultural Heritage", 2010, <http://www.mediageo.it/ojs/index.php/archeomatica/article/view/177>
- [Pre10] Presley A., "Pantshop Pro vs. Photoshop", 2010, <http://www.brighthub.com/internet/web-development/articles/30727.aspx>
- [Rem10] Remondino F., "Rilevare e rendere visibile l'invisibile. Documentazione 3D e multispettrale per fruizione di capolavori artistici", 2010, <http://www.mediageo.it/ojs/index.php/archeomatica/article/view/163>
- [Sev12] Sevo D., "History of Computer Graphics", 2012, [http://www.danielsevo.com/hocg/hocg\\_1960.htm](http://www.danielsevo.com/hocg/hocg_1960.htm)
- [Spa07] Spallone R., "3D Digital Modelling as a Method for the Reconstruction of the Historical Image of the City: the Case of Piazza Bodoni in Turin (Italy) at the end of Nineteenth Century", 2007, <http://www.isprs.org/proceedings/xxxvi/5-c53/papers/FP134.pdf>
- [TAB10] Tucci G., Algostino F., Bonora V., Bucalossi L., Cini D., Conti A., Fiorini L., Nobile A., Menci L., Ceccaroni F., Bianchini D., Ghezzi M., "Musealizzazione virtuale. Esperienze di rilievo e modellazione 3D per un allestimento interattivo ed accessibile da Web", 2010, <http://www.mediageo.it/ojs/index.php/archeomatica/article/view/167>
- [Tak12] Takahashi D., "Game developers, start your Unity 3D engines (Interview)", 2012, <http://venturebeat.com/2012/11/02/game-developers-start-your-unity-3d-engines-interview/>
- [Tak14] Takahashi D., "John Riccitiello sets out to identify the engine of growth for Unity Technologies (interview)", 2014, <http://venturebeat.com/2014/10/23/john-riccitiello-sets-out-to-identify-the-engine-of-growth-for-unity-technologies-interview/>
- [Ver07] Barbara Vernia, "Forlimpopoli - Rocca Albornoziata", 2007, <http://www.emiliaromagna.beniculturali.it/index.php?it/108/ricerca-itinerari/16/51>

## ALTRO

- [NG13] Animation Masterclass: Nierva R., Gordon A., "Story / Character / Animation", 2013

## IMMAGINI

<http://www.fotografieitalia.it/foto/2578/2578-07-33-45-7922.jpg>