On-Screen Visual Balance Inspired by Real Movies

Cunka Sanokho and **Marc Christie**
INRIA - University of Rennes 1
Campus de Beaulieu
Rennes, France

Abstract

The computation of appropriately balanced shots in synthetic animation movies is crucial both to properly convey the content and meaning of a 3D scene, and to conform to classical aesthetic criteria. Visual balance in shots can be defined as the arrangement of pictorial elements to form a unified harmonious whole in a picture. Current techniques used in automated viewpoint computation typically encode a set of established rules and conventions from literature in photography and cinematography into metrics to assess the quality of on-screen composition and balance. In this paper, we move beyond these representations by proposing an elaborate model of on-screen visual balance that learns from annotated shots in real movies. The model includes features such as size, silhouette, position and luminance of target objects, together with metrics related to actors' positions, orientations and gaze. We then show how the model relies on data from real shots to evaluate balance in synthetic shots.

Introduction

Aesthetics is a central criteria when considering the quality of shots in both real and virtual environments. Proposing reliable and automated techniques to estimate and to enforce aesthetics in real or synthetic shots is a challenging task due to the difficulty of defining and formalizing aesthetics in a reliable and efficient computational model. Despite the difficulty, some aspects of aesthetics have been addressed in both image analysis and computer graphics communities. In image analysis, the objective is to evaluate the quality of a real shot (photography) and propose modifications in its framing (Liu et al. 2010) or content (Zhang, Wang, and Hu 2013) that conform to common aesthetic conventions. In computer graphics, the objective is to search for the best shot satisfying such conventions (Bares et al. 2000; Abdullah et al. 2011) by changing the viewpoint or the location of objects in the scene to modify the shot layout.

Aesthetics is strongly dependent on the appropriate composition of pictorial elements on the screen, and visual balance plays a key role in this composition (Mascelli 1965). Balance represents the equilibrium of visual weights in the screen, *i.e.* equilibrium of the visual interests one perceives.

This aspect has been relatively under-addressed. Modeling the sense of balance in shots requires to compute the weight (or leverage) of each visual feature in the global picture balance, including luminance, actors positions, silhouettes, size on the screen, eyes position of actors and gaze. Currently most contributions in the domain rely on a simplified model of balance (location and area of visual entities on the screen in (Swanson, Escoffery, and Jhala 2012; Bares 2006) or in (Zhang, Wang, and Hu 2013)).

In this work, we propose an elaborate model of balance based on multiple visual features. Our hypothesis is that each of these features carries a part of an object visual weight and that the aggregation of these weights in the screen provides the sense of balance, *i.e.* an equilibrium of visual weights. The principle of our model consists in evaluating the visual balance of synthetic shots by learning from a selection of well balanced shots extracted from real movies using a *balance feature space*. Given that the general balance estimation problem is strongly related to the semantics of entities on the screen, we restrict our analysis to the subset of shots only containing actors.

Given that balance is a complex combination of these features, we first propose to use a collection of annotated shots from real movies that we consider properly balanced, to assess the relative importance between the features in each image. The balance assessment for these pictures is done by an expert who grades them with a binary value: balanced or not balanced. This provides us with a feature-space in which we can position each annotated shot with its particular combination of weights. In a second step, we use this feature-space to propose a distance metric between shots in terms of balance. Any new shot can be positioned in this feature-space and its distance to existing shots can be measured using a weighted combination of the nearest annotated shots. This distance metric can therefore be used to evaluate the balance of the new shot. Results show that the technique effectively characterizes well-balanced shots with one and two actors.

In its implementation, our method does not rely on shot similarities in terms of content but on more low level features such as luminance, silhouette, eye position and gaze. Our approach finds applications in automated viewpoint computation and automated cinematography which are receiving a strong interest given the complexity and visual quality of nowadays 3D environments, and the necessity of conveying such contents.

Related Work

Automated computation of viewpoints in 3D environments started to receive a strong interest in the computer graphics community given the increase in amount and quality of 3D models and the necessity to properly convey these 3D contents. The objectives of automated viewpoint computation techniques are strongly pertained to the targeted application. For example in medical visualization, the techniques will help to focus on a specific organ of a body, ensuring its visibility and relation to other organs in a focus-plus-context approach (Mühler et al. 2007). In information visualization, appropriate viewpoint computation supports for example the exploration of historical data (Stoev and Strasser 2002).

Viewpoint computation has generally been expressed in terms of viewpoint entropy, which is a measure of the information provided by a point of view (Vázquez et al. 2001). Viewpoint computation therefore aims at optimizing the viewpoint to maximize the amount of information it conveys and has been well addressed by evaluating visibility, projected surface or complexity of the silhouette (Vieira et al. 2009),(Vázquez et al. 2003). However, the relevant display of information from a viewpoint also relies on the proper layout of visual elements on the screen (referred to as visual composition or simply composition).

Approaches that tackle viewpoint composition generally encode classical photographic composition rules and cast the problem as an optimization one: composition rules are expressed as cost functions over the camera parameters, and the optimization technique searches within the camera parameters for the viewpoint minimizing the aggregated cost functions. For example, Olivier *et al.* (Olivier et al. 1999) propose an extended composition language (enforcing the location, orientation and size of an object on the screen) and adopt a meta-heuristic search algorithm (namely a genetic algorithm due to the presence of many local minima). Other approaches follow a similar principle with variations in the language and algorithms (Bares et al. 2000)(Jardillier and Languénou 1998). Issues are related to the necessary abstraction of complex objects as points or simple primitives on the screen for means of efficiency in computations.

Visual balance in composition, defined as the arrangement of pictorial elements to form a unified harmonious whole, has been relatively under-addressed due to the difficulty in formalizing and therefore evaluating balance. Interestingly, in the domain of image analysis, multiple metrics have been defined to re-frame or re-compose existing pictures (Zhang, Wang, and Hu 2013). Typically Ligang *et al.* and Zhang *et al.* simply represent the notion of balance as the distance from the center of mass of all objects to the image center (Liu et al. 2010)(Zhang, Wang, and Hu 2013). The intrinsic complexity of formally defining visual composition and balance has pushed the research community to consider learning techniques from a database of annotated shots, and our approach follows this trend. In (Swanson, Escoffery, and Jhala 2012), the authors rely on a data collection with crowd-sourcing annotation to study visual composition preferences, using metrics of balance, thirds alignment, symmetry and spacing. However, the definition of metrics to estimate balance requires to properly define the mass of objects using their luminance, silhouette, size and position on screen. Furthermore mass depends on the nature and semantics of the targets.

This paper proposes a formalization of visual balance in composition by restricting the problem to viewpoints with actors for which the semantics are well defined. To estimate balance in shots, we rely on the annotation of real shots from movies to construct a feature space in which the feature distance to a real shot characterizes its degree of balance.

Overview

Our overall process encompasses two stages (see Figure 1). The first stage consists in extracting the features involved in visual balance. In section *Visual Features*, we show how the features are extracted and annotated from real movies in order to create a feature space that describes the best feature associations to create well balanced shots. Although this feature space provides us with a collection of feature vectors for each shot, it does not carry any information on the relative importance of the features of one balanced shot.

In the second stage, we measure the relative importance of these features by setting up a theoretical center of mass located in the center of each picture. An optimization method is used to compute the leverage value associated to each feature such that the computed center of mass is the same as the theoretical center of mass. These leverage values are weights that reflect the influence of the feature in the balance value. The position of the center of mass is expressed as a weighted linear combination of the features. The process is described in section *Optimization*. Our hypothesis is therefore that the association of good features with the correct leverage values provides well balanced shot. We can then assess balance in a new shot by comparing its features with the closest balanced shots in our *balance feature space*. We compute the new center of mass by interpolating the leverage vectors associated with those balanced shots as described in section *Evaluation of balance in synthetic shots*. Finally, the degree of balance is given by the distance between the new center of mass and the theoretical center of mass (center of the screen).

Visual Features

Following Mascelli (Mascelli 1965), we propose to define visual balance in a picture as a combination of the visual weights of the entities it encompasses. In our context, visual weight is influenced by 2-dimensional features on the actors and on their screen positions. We referred to general recommendations in literature to propose a list of features that were considered significant in visual balance (see Table 1). A semantic level was introduced that distinguishes the main actor from secondary and auxiliary actors (see Figure 2). The main actor performs the action. For example if the current action is "A talks to B" actor A will be considered as the main actor in the shot. The secondary actor is the one who participates in the action. For example if the current ac-

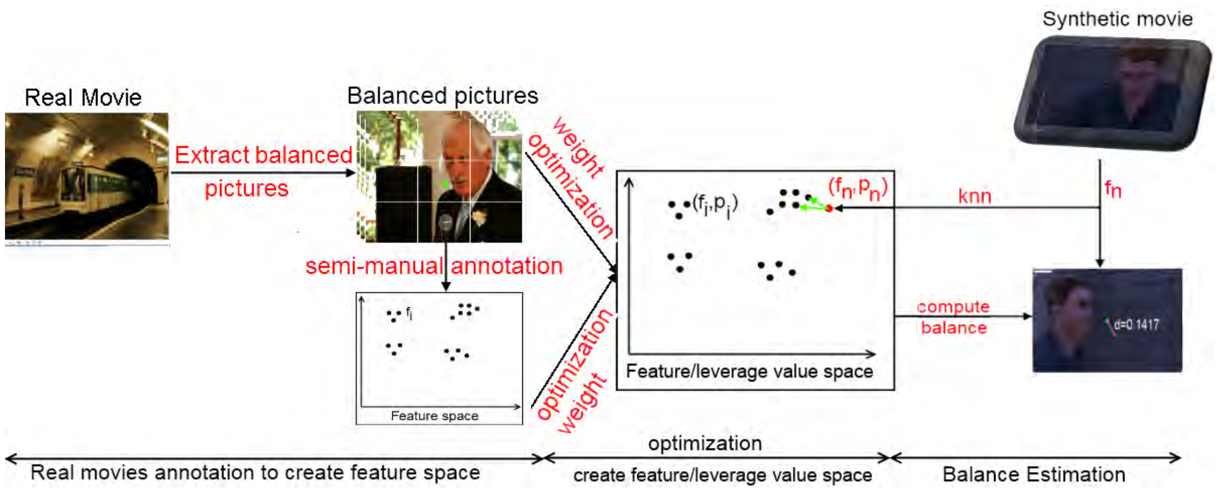

Figure 1: Estimating and correcting visual balance in synthetic shots.

Feature	Symbol
actor's luminance	L
actor's isolation	Is
actor's sizes in the 2D picture	O
actor's on-screen position	P
main actor's gaze direction	Ed
main actor's body orientation	Bd
main actor's head direction	Hd
main actor's face size	Hs
main actor's head center	$Hp_{x,y}$
picture intrinsic areas masses	Ma
actor's importance in the action	Imp
actor's eye i position in height and width	Ep_{ix}

Table 1: Significant visual features in terms of visual weight and their symbols.

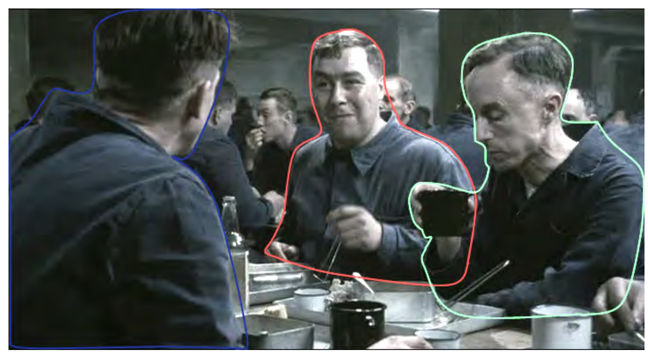

Figure 2: Actor's importance: the red actor is the Main actor. He is talking to the blue actor who is the secondary actor. The green actor is an auxiliary actor as he does not take part in the current action: he can be replaced by any other actor but participates in the balance.

tion is "A talks to B", B is the secondary actor. The auxiliary actors do not participate in the current action. They can be performing background tasks which may be essential to understanding the current action and participate in the balance.

Image balance results from a combination of these features. To be able to quantify balance in images, we propose to extract all these features from real shots taken from movies and approved by an expert. In the following paragraphs, we will explain how these features are extracted and the post-processing we perform on the extracted data.

Extraction

Areas' masses extraction The principle behind this feature is that the screen is divided in multiple areas. Some areas in the screen are more attractive to the eye than others. Thus, the location of pictorial elements on the screen influences their weight. For example (Mascelli 1965) underlines that an object placed in the center of the image has less weight than an object placed away from the center. A direct consequence of this is that a heavy object should be closer to the center whereas a lighter object should be away from the center. The areas' masses we introduce here are weight attenuation factors applicable to an object when positioned at a certain location on the screen. The areas' masses are to be learnt during the optimization step for every picture. Figure 3 displays an example of area masses.

Luminance Light is an important component in pictorial unity (Mascelli 1965). Well illuminated actors tend to have a more important weight in pictorial balance as they attract the observer's eye. In our approach, an actor's mean luminance is normalized with respect to the entire shot average luminance. Luminance is computed from the RGB values by converting into XYZ space where Y is the luminance. An actor's luminance is computed with the following equation:

Figure 3: Example of average areas' masses distribution on the picture. The values are normalized in the interval [0,1]. The dark areas will apply less attenuation to the objects (e.g. provide more weight to objects) whereas brighter areas will apply more attenuation (e.g. provide less weight to objects). In this example, we divided our picture in 32 (8x4) areas.

$$L_i = \frac{\sum_{w=0}^{height} \sum_{w=0}^{width} \begin{cases} L(I(h,w)) & if\ I(h,w) \in A \\ 0 & else \end{cases}}{\sum_{w=0}^{height} \sum_{w=0}^{width} L(I(h,w))} \tag{1}$$

With:

- I, the image we process,
- h, w, the coordinates of a pixel in the image,
- A, the actor being processed in the image,
- $height$, $width$, the size of the image,
- $L(I(h,w))$ the luminance computed at pixel I(h,w).

Figure 4 shows luminance values in function of the actor's importance.

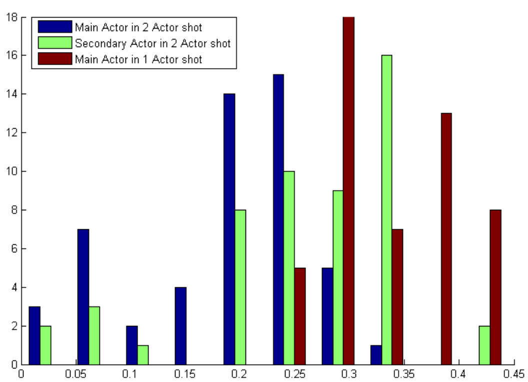

Figure 4: Distribution of actors' relative luminance (ratio between the actors' luminance and the entire shot luminance on the x-axis). Main actors appear to be better illuminated than secondary actors. In addition, on one actor shots, the ratio of mean actor luminance on mean shot luminance is always higher than 0.25.

Isolation It is admitted (Mascelli 1965) that stacked objects have less impact on eye attraction. Indeed it is hard to separate the object of interest from distracting objects. The rule proposed by Mascelli is to say that stacked objects would have less weight than isolated objects. In our framework we use a boolean value to describe isolation. We made this choice because isolation is hard to define. It is either a distance in 2D between objects or the distance in 3D. This value highly depends on the operator's appreciation. For the most, we selected pictures with good isolation (e.g. , the actors are fully visible and not crowded with distracting actors or object). Isolation value is 1 for good isolation and 0 for bad isolation.

Importance of the actor The central actor of interest of the picture is encoded by three values, Main = 2, Secondary = 1, Auxiliary = 0. The main actor generally attracts the most visual attention.

On-screen size and position This is one of the most important features as it provides information on camera angles and on actor size projected on the screen. Alone, size significantly contributes to balance but a director can balance big objects with small ones. Since big objects are more important in terms of visual weight, balancing small objects should have other properties as better illumination, position or movement to compensate. Size is represented as the ratio of the number of pixels covered by the actor and the image size in pixel. We also save the projected pixels of the actor in the 2D picture to access the on-screen position. Figure 5 shows the distribution of on-screen positions of one actor shots and Figure 6 displays main and secondary actors on-screen positions in two actors shots.

Figure 5: Distribution of positions in a one actor shot. The green part is the area where there are less than 33% of presences, the blue one is for less than 66% and the red one is for more than 66%.

Head center, Eyes position and Head Size The Head center depicts the actors head position. It is an important positional feature since in good compositions, the actor's head ought to be located around the image's more powerful points (for example thanks to rule of thirds). In a similar way, eyes position are essential in balance as they are also often located on the image more powerful points (Mascelli 1965).

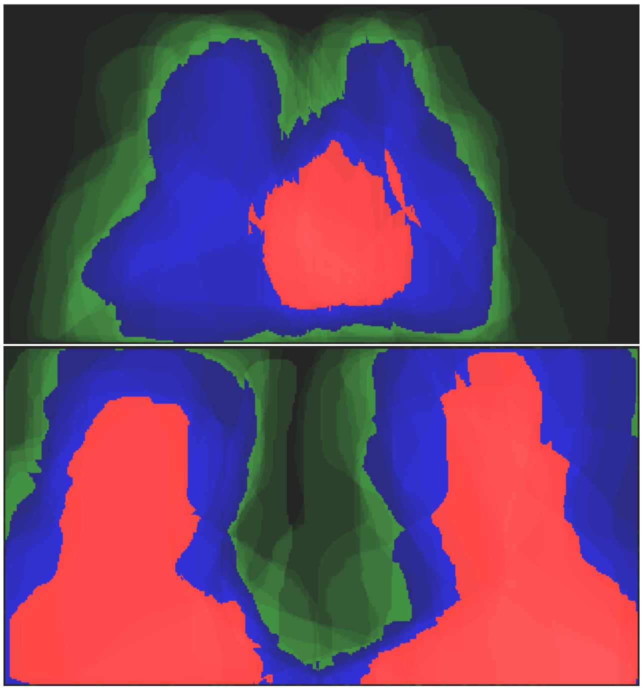

Figure 6: Distribution of positions in two actors shots. The upper picture is for the Main actor and the lower one is for the Secondary actor. The green part is the area where there are less than 33% of the actors, the blue one is for less than 66% and the red one is for more than 66%.

Combined to the actors head size, they provide information on shot length (close up, long,...).

Gaze and Head direction The gaze direction influences image balance as it guides the spectator's eyes through the scene and adds weight to the objects or directions targeted by the actor's gaze. It can differ from the orientation of the head (see shot 8 in Figure 8). Gaze and head direction are computed using the same principle. Their value is the angle between the vertical of the picture and a vector representing the direction of gaze or head orientation. This value is scaled to the interval $[0, 1]$. The gaze is oriented toward the virtual location of the action's protagonist (it can be an off-screen object, see Figure 7). The head direction is oriented in the direction faced by the actor on the picture.

Body orientation Body orientation is the estimated orientation of the actors body on the screen. We represent orientation as a scalar value according to the rules defined in Figure 8.

Feature Space

As a result, we have obtained a collection of over 100 annotated shots for one-actor and two actors shots. The shots are organized in a feature space where each shot is represented by its extracted features. All the extraction was performed manually with a software we developed. For example to determine actor position we performed manual delineation of the actor and used a region growing segmentation algorithm

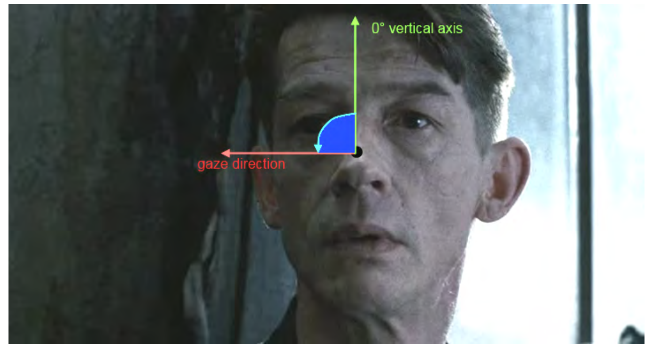

Figure 7: The actor's head seems centered but the gaze direction goes from right to left. The value we use is the blue angle between the vertical direction and the gaze direction.

Figure 8: Example of body orientation: in shot 1 the actor is seen from front which corresponds to camera position 1 in sub-picture 9. Pictures 2 to 8 correspond respectively to position 2 to 8 in sub-picture 9. These positions are normalized in the interval $[0, 1]$.

to extract the projection. All the positional features are normalized by the picture size.

Optimization

By considering that all the annotated shots in our collection are well-balanced shots, we formulate the hypothesis that there exists a virtual balance point located in the center of these shots (in a way, the center of balance in the image or theoretical center of mass). We define this theoretical center of mass as a linear combination of the visual weights relative to the extracted features. However, we need to assess the relative contribution of each feature in the final balance. To this end, we associate a leverage value to each feature. This leverage value actually represents the contribution of the feature to balance. We refer to this value as a leverage value and not as a weight to avoid confusion with visual weights. We then rely on an optimization process to compute, for each image (shot), the set of leverage values such that the theoretical center of mass is located at the center of the screen. This process provides us with a feature/leverage value space that will be used later to assess and correct balance in pictures. To compute these leverage value, we follow a three stages process: first, definition of features' domain of application or scope, then definition of the cost function which value needs to be optimized, eventually optimization of the leverage value set for each image.

Features scope

We distinguish two types of features: those which are relative on the actor and those which are applied on a part of the image. The first category includes on-screen size, luminance, head size, isolation, center of interest and body direction for all actors. We normalize their values according to the actor size in pixels to account for their relative importance. Head direction and eyes direction are applied to a specific area of the picture. This area position and size depends of their value. The algorithm 1 describes the method to compute these areas. The algorithm is given for head direction but the exact same rules are applied for eyes direction.

Algorithm 1: Head direction : decides if a the head direction is defined for the pixel x.

Input: Hp_x : The head position.
Input: Hd : The head direction normalized in $[0, 1]$.
Input: x : a pixel width coordinate.
Output: $Hdir =$
$$\begin{cases} 1 & \text{if} \quad \textit{HD applicable to x} \\ 0 & \text{otherwise} \end{cases}$$

```
if Hd < 0.45 ∧ Hd > 0.05 then
    if x < Hp_x then
        Hdir ← 1;
    else
        Hdir ← 0
    end
end
if Hd > 0.55 ∧ Hd < 0.95 then
    if x > HPX then
        Hdir ← 1;
    else
        Hdir ← 0
    end
end
return Hdir
```

Cost function

The cost function to be optimized is expressed as a linear combination of the features. Their leverage values are combined with the areas masses presented in Figure 3. The area masses are not known before the optimization, they are learnt by the optimization method. In the following we will explain the optimization process for one picture. For a given picture, a weight is associated to each actor, this weight represents the perceptual impact of the actor on the spectator. We consider this weight as linear combination or various features presented in section *Visual Features*. Let $feat_i$ be a vector of features associated to an actor

$$f = [O, L, Hd, Ed, Hs, Ep_{ix}, Is, Hp_x, Imp, Bd, Rs] \tag{2}$$

and $param_i$ a vector of the same size containing the leverage values of $feat$. The weight W_i of the actor i is given by:

$$w_i = f_i * p_i \tag{3}$$

for each area of the screen W_z the weight is given by the union of the weight of all the actors on the area, weighted by the mass of the area.

$$W_z = Ma_z * \sum\nolimits_{i(\textit{if object } i \in area)} (w_i) \tag{4}$$

with : $\sum_z W_z = 10$.

The current center of mass is given by (assuming our image's coordinates range from $(0, 0)$ to $(1, 1)$):

$$B_r = \sum_{i=0}^{1} \sum_{j=0}^{1} W_{i,j} \begin{bmatrix} i \\ j \end{bmatrix} \tag{5}$$

Our optimization method consists in minimizing the distance between the current center of mass and the theoretical center of mass. Finally our cost function is:

$$argmin_f ||Br - Br_t|| \tag{6}$$

with : Br_t the theoretical center of mass.

The algorithm 2 describes the way we compute our cost function. In addition to the extracted features, the input variables are the following : Rs : the $\frac{Hs}{O}$ ratio, I : the input image, Br : the current center of mass, Br_t : the theoretical center of mass.

Algorithm 2: Balance function

Input: $O, L, Hd, Ed, Hs, Ep_{ix}, Is$.
Input: $Hp_x, Imp, Bd, Rs, p, Ma, P$.
Output: $cost$
$const = p_o * O + p_l * L + p_{hs} * Hs + p_{Imp} * COI + p_{is} * Is + p_{rs} * Rs + p_{bd} * Bd$
$Br := [0, 0]$
$Br_t := [0.5, 0.5]$

for $h := 1$ ***to*** $Height(I)$ **do**
  for $w := 1$ ***to*** $Width(I)$ **do**
    $Mass \leftarrow Ma(h, w)$
    $I_{hw} \leftarrow \begin{cases} I(h, w) & if\ I(h, w) \in P \\ 0 & else \end{cases}$
    $Var \leftarrow Hd * x(3) + x(4) * Ed + x(8) * HP + x(6) * E_{ix}$ `/* where they apply */`
    $W_z \leftarrow Mass * (I_{hw} * const + Var)$
    $Br \leftarrow Br + W_z * [h/Height(P), w/Width(P)]$
  end
end
return $||Br - Br_t||$

The process is the same for two actors except that new features are added: the second actor's on-screen size, brightness, isolation, importance, and position.

Optimization in practice

We used a constrained nonlinear optimization technique to solve the set of leverage values for each image. The first constraint is to set the sum of all parameterized features to 10. The second constraint is to set the sum of all screen positions to 10. We made these choices to make sure that the interval for our leverage values is wide enough to represent them all and to reach a solution. We computed 132 leverage values for one actor and 136 for 2 actors. Among those, 120 were screen position values. In a more practical way, we used the Matlab's fmincon interior-point minimization algorithm to compute the leverage values. We limited our objective function to 10^{-5} and we set the maximum number of iterations to 20000.

Evaluation of balance in synthetic shots

Interestingly, our approach to estimate balance can be used on synthetic shots. Indeed, given that all the information of a 3D environment is known before hand, we can automatically extract the features needed in the computation. One characteristic of our algorithm is that it does not rely on pictorial high level similarities but on low level features similarities. To compute the leverage values vector for our image, its feature is projected on the feature space. An inter-feature distance is used to determine which images are the closest in term of extracted features. We use a k nearest neighbors (k-nn) algorithm to perform this task. The set of leverage values of our synthetic shot is computed by performing an interpolation on the leverage values associated to the k-nn.
The current center of mass is computed with the extracted features, and the computed leverage values. Here we consider that if the center of balance is in a close area around the center of the screen, the shot is balanced ($d < 0.1$). In the following, we present some of our results for one actor shots and for two actors shots.

One actor shots

We performed our extraction over a dataset of 50 shots. Each shot was divided in 120 areas of constant area's mass: 12 along the height and 10 along the width of the picture. This choice was made because a bigger resolution would not be relevant to describe our pictures. We then applied our method to assess the balance of synthetic shot extracted from an automated viewpoint computation tool (Lino et al. 2011)). Figure 9 shows an example for one actor in random camera positioning.

Two actors shots

We performed our experiment in the same conditions than previously for two actors. We noticed about a one hour difference in computation time between one and two actors for one shot. Figure 10 shows an example for two actors in random camera positioning.

Figure 11 and Figure 12 show examples of balance assessment in different configurations for one actor and two actors shots.

Figure 9: Composition evaluation for a one actor shot. Top left picture represents the shot to evaluate. Three best matches were computed for this picture's configuration in feature space. The computed center of mass is here represented by the red point and the center of the picture by the green point, showing a significant distance to a correct composition.

Figure 10: Example of composition evaluation on a two actor shot. We can notice in this example that the main actors are away from the center. This cause a disequilibrium. When we assess balance for this picture we can see that the computed center of mass is far away from the center.

Conclusions and future work

We present a novel approach to evaluate balance in synthetic shots, by proposing an evolved metric of balance. Existing techniques only consider balance from the point of view of the surface and location of targets on the screen, while art literature in the field provides a more general definition based on the equilibrium of visual weights. The results we present show that our method can more precisely assess image balance by considering features such as gaze direction, on-screen actor position and orientation, luminance or layout of masses in the picture. Onoing work relies on our balance estimation technique to automatically correct balance in synthetic shots through appropriate camera displacements. We also will consider constraints due to gaze continuity and spatial continuity when computing sequences of balanced synthetic shots.

Figure 11: These examples show one actor situations. In the upper pictures we can see that disequilibrium is caused by the nose room and the excessive luminance of the door with respect to the actor. In the second picture in particular, the size of the actor increases the disequilibrium. The lower two pictures are well-balanced examples with correct sizes of actors. We can see that the luminance of the scene has an influence in the overall balance of the picture.

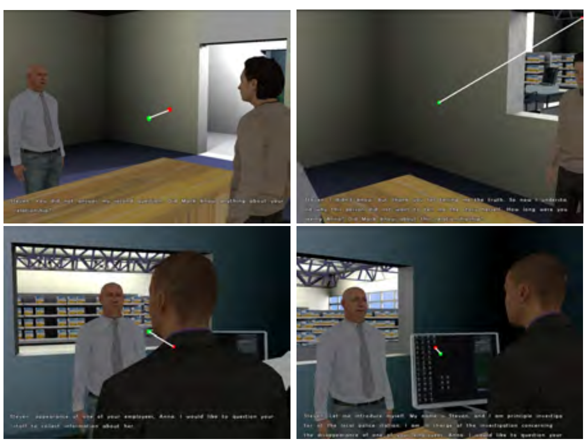

Figure 12: These examples show two actors situations. The upper example shows the same scene from two different angles. In the first picture the actors are out of frame and the center of mass is pulled by the right bright area. In the second, the framing is better but still the luminance causes a small disequilibrium. The lower example shows the same kind of situation. The difference of balance here is linked to the camera angle that modifies the actors' position in 2D.

References

Abdullah, R.; Christie, M.; Schofield, G.; Lino, C.; and Olivier, P. 2011. Advanced composition in virtual camera control. In etal., L. D., ed., *Smart Graphics*, volume 6815 of *Lecture Notes in Computer Science*, 13–24. Springer.

Bares, W.; McDermott, S.; Boudreaux, C.; and Thainimit, S. 2000. Virtual 3D camera composition from frame constraints. In *Proceedings of ACM Multimedia)*, 177–186. ACM Press.

Bares, W. H. 2006. A photographic composition assistant for intelligent virtual 3d camera systems. In Butz, A.; Fisher, B. D.; Krüger, A.; and Olivier, P., eds., *Smart Graphics*, volume 4073 of *Lecture Notes in Computer Science*, 172–183. Springer.

Jardillier, F., and Languénou, E. 1998. Screen-space constraints for camera movements: the virtual cameraman. In *Proceedings of the Eurographics Conference (EG 98)*, volume 17, 175–186. Blackwell Publishers.

Lino, C.; Christie, M.; Ranon, R.; and Bares, W. 2011. The director's lens: an intelligent assistant for virtual cinematography. In *Proceedings of the 19th ACM international conference on Multimedia*, MM '11, 323–332. New York, NY, USA: ACM.

Liu, L.; Chen, R.; Wolf, L.; and Cohen-Or, D. 2010. Optimizing photo composition. *Comput. Graph. Forum* 29(2):469–478.

Mascelli, J. 1965. *The Five C's of Cinematography: Motion Picture Filming Techniques*. Cine/Grafic Publications, Hollywood.

Mühler, K.; Neugebauer, M.; Tietjen, C.; and Preim, B. 2007. Viewpoint selection for intervention planning. In Museth, K.; Möller, T.; and Ynnerman, A., eds., *EuroVis*, 267–274. Eurographics Association.

Olivier, P.; Halper, N.; Pickering, J.; and Luna, P. 1999. Visual Composition as Optimisation. In *AISB Symposium on AI and Creativity in Entertainment and Visual Art*, 22–30.

Stoev, S. L., and Strasser, W. 2002. A case study on automatic camera placement and motion for visualizing historical data. In *VIS '02: Proceedings of the conference on Visualization '02*, 545–548.

Swanson, R.; Escoffery, D.; and Jhala, A. 2012. Learning visual composition preferences from an annotated corpus generated through gameplay. In *CIG*, 363–370. IEEE.

Vázquez, P.-P.; Feixas, M.; Sbert, M.; and Heidrich, W. 2001. Viewpoint selection using viewpoint entropy. In *VMV '01: Proceedings of the Vision Modeling and Visualization Conference 2001*, 273–280. Aka GmbH.

Vázquez, P.-P.; Feixas, M.; Sbert, M.; and Heidrich, W. 2003. Automatic view selection using viewpoint entropy and its application to image-based modelling. *Computer Graphics Forum* 22(4):689–700.

Vieira, T.; Bordignon, A.; Peixoto, A.; Tavares, G.; Lopes, H.; Velho, L.; and Lewiner, T. 2009. Learning good views through intelligent galleries. *Computer Graphics Forum* 28(2):717–726.

Zhang, F.-L.; Wang, M.; and Hu, S.-M. 2013. Aesthetic image enhancement by dependence-aware object recomposition. *IEEE Transactions on Multimedia* 28(2).