

両眼視差方式の原理に基づく 「快適な立体視」を リアルタイムゲームへ適用するには

バンダイナムコゲームス
開発スタジオ アドバンステクノロジーディビジョン
技術部 開発サポート課 プログラマ 石井源久

※本スライドに記載されている会社名や商品名は、
該当する各社の商標または登録商標です。本文中
では、®、™マークを記載していません。

社名表記

- 株式会社バンダイナムコゲームス (日本語表記)
- NAMCO BANDAI Games Inc. (英語表記)
- NBGI (公式の略称)

はじめに



- 謝辞
- 業務紹介
- 「3DC安全ガイドライン」について
- 今回の講演テーマについて

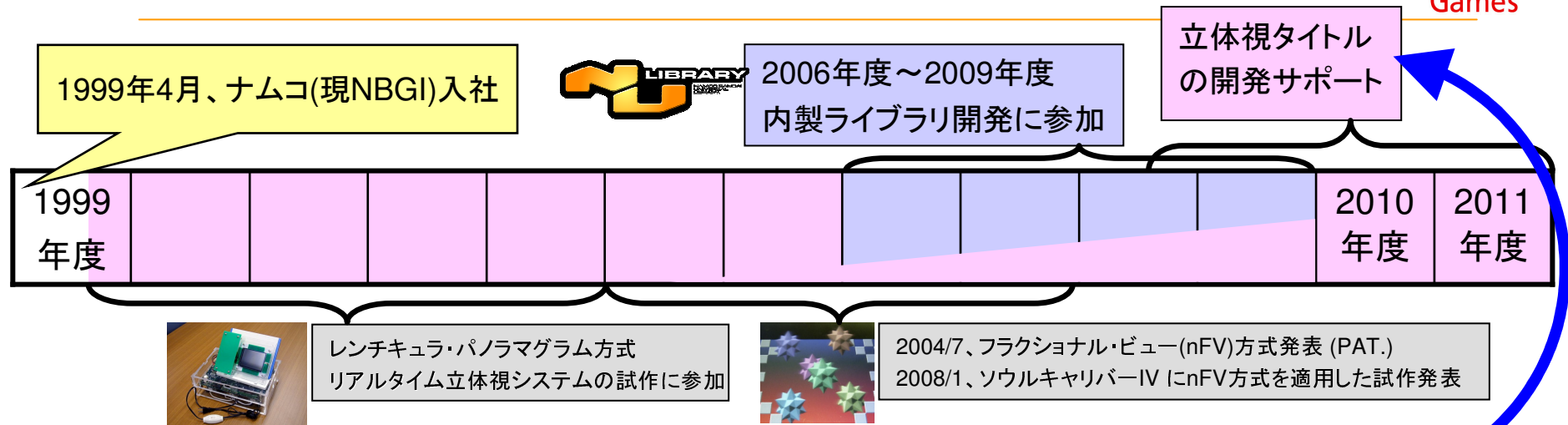
謝辞



本講演にあたり、以下の文献、セミナーの内容を参考にいたしました。また、記載した以外にも多くの専門家の方々よりご教示を賜りました。ここに感謝申し上げます。

1. 3DC安全ガイドライン 2010年4月20日改訂版, 3Dコンソーシアム
2. 3D立体映像表現の基礎-基本原理から製作技術まで, 河合隆史, 盛川浩志, 太田啓路, 阿部信明, オーム社, 2010/9/20
3. SIGGRAPH ASIA 2009, The Look of “Up”: A Filmmaker’s Guide to the Pixar Process, Colin Thompson, Colin Thompson, Thomas Jordan, Patrick Lin (Pixar Animation Studios), 2009/12/16
4. CEDEC 2010, 「良い」3D立体視ゲームを作るためのプログラミングとデザイン入門, 福本正紀, 大戸友博 (株式会社ソニー・コンピュータエンタテインメント), 2010/9/1
5. CEDEC 2010, 3D映像制作:ハリウッドの現場から, Buzz Hays(Sony Corporation of America), 2010/9/2
6. CEATEC 2010パネルディスカッション, 3D映像への期待と生体安全性について, 板東武彦, 黒岩 義之, 不二門尚, 山家智之, 千葉滋, CEATEC2010, 2010/10/5
7. Digital Contents Expo 国際3D Fair 2010, 3Dエルゴノミクスセミナー, 河合隆史, 2010/10/14
8. InterBEE 2010, 3Dコンテンツ“人間の生体への安全性確保”に向けて, 千葉滋, 田中勉, 畑田豊彦, 矢野澄男, 板東武彦, 伊藤崇之, 川上一郎, 2010/11/19
9. 立体Expo2010, 「立体映像の生体安全性と視覚的適性」～3D映像がもたらす人への効能～, 半田知也, 2010/12/8

業務紹介(石井)



2010年度から、**立体視タイトルの開発サポート**がメイン業務。

製品		開発中
リッジレーサー3D (Nintendo 3DS) 2011/2/26, 3.5インチ	マキシマムヒート (アーケード) 2011/3, 32インチ	デッドストームパイレーツ3D(仮) (AMショー2010) 2010/9/9-11, 150インチ

- 立体感の設定方法の説明、表現技法の説明
- 社内 “S-3D” 資料作成(プログラミング、表現技法、ガイドライン)
- 問題解決

S-3D!

GL-p10

他のNBGI発売タイトル

- (1) サンダーセプターII (アーケード, 1986/11)
:
(2009年以降)
(2) Invincible Tiger - The Legend of HANTAO (Xbox360 Live Arcade, 2009/8, NBGA)
(3) リッジレーサー7 - 3D License Ver. (PS3, 有料ダウンロードパッチ, 2010/10)
(4) Enslaved - Odyssey to the West
(PS3/Xbox360, 有料ダウンロードコンテンツ購入特典, 2010/12, NBGA)
(5) 太鼓の達人 (SoftBank 005SH, プリインストールソフト, 2011/2/25)
(6) GUNDAM THE 3D BATTLE (3DS, 2011/3/24)
(7) プロ野球 ファミスタ2011 (3DS, 2011/3/31)

- プレイヤーに周知すべき内容の表記や、デバッグ方法の策定に協力。
- 製品の結果を次期製品の品質向上にフィードバック。

S-3D!

(「3DC安全ガイドライン」の該当ページ)

「3DC安全ガイドライン」について

- 「3Dコンソーシアム」<http://www.3dc.gr.jp/jp/index.html>が策定したガイドライン(2010年4月20日改訂版)。
「3D産業の健全な育成を図るべく一般公開」されており、以下の内容を含む。
 - 視聴者に周知すべきガイドライン
 - コンテンツ製作者のためのガイドライン
 - 製造者のためのガイドライン
- 立体感には個人差が大きい。
 - 映像製作者も例外ではない。
 - 多くの人に快適に見ていただくことを前提とした立体感の設定には、客観的に判断するための指標が必要。

そこで、多くの文献や調査結果に基づく、
「3DC安全ガイドライン」
を良く理解し、活用することが望まれる。

今回の講演テーマについて

両眼視差方式の原理に基づく
『快適な立体視』
をリアルタイムゲームへ適用するには

(1)
両眼視差
方式

(2)
快適な
立体視

(3)
リアルタイム
ゲームへ
適用

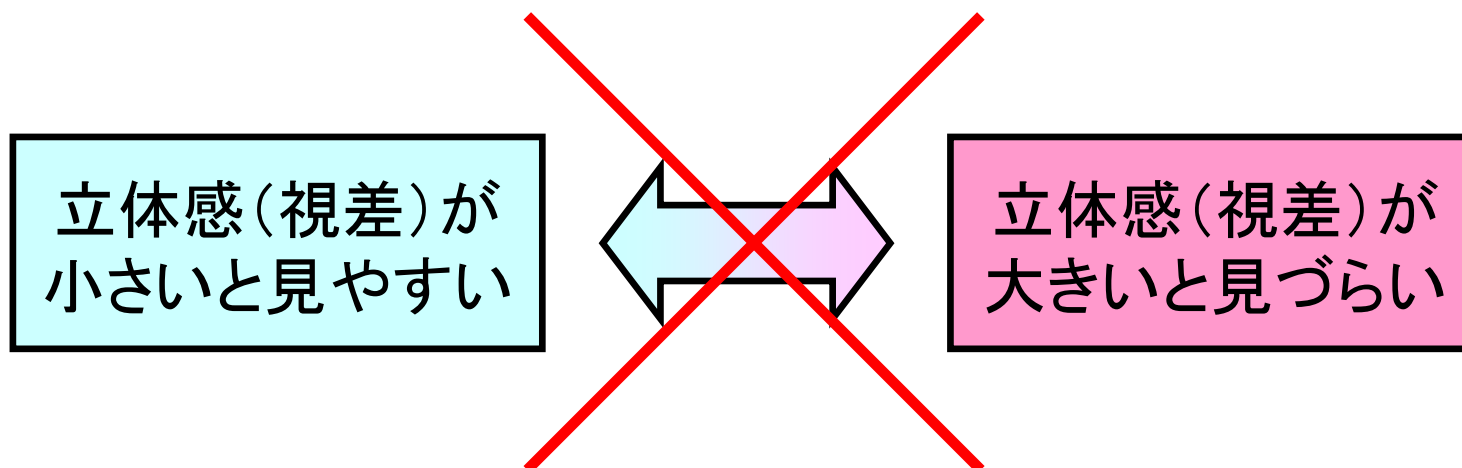
まず、これらのテーマを簡単に。

「両眼視差方式」

- 「3DC安全ガイドライン」は、「両眼視差方式」での映像制作を前提にしている。
- 現在普及している、立体視ゲームのプラットフォームは、メガネ式・裸眼式を問わず、「両眼視差方式」を前提にしたものである。
- (両眼)視差： 多くの人にとって、
立体感の最も強力な手がかり
- 視差の付け方(ステレオカメラワーク)が重要

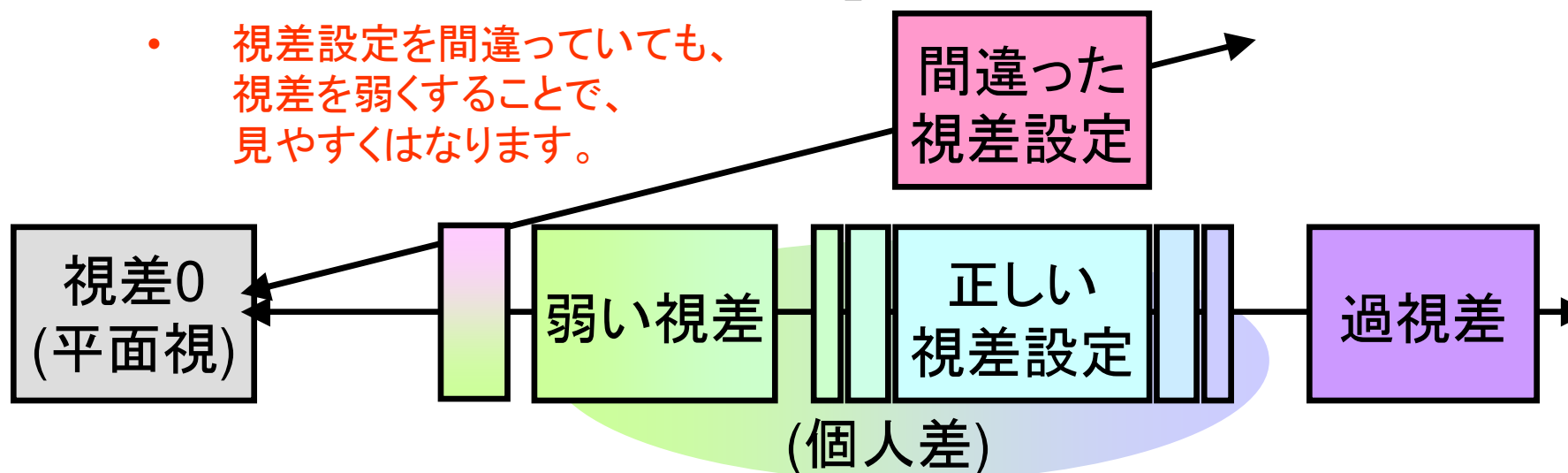
「快適な立体視」

- 「見やすい」と「立体感がある」は、必ずしも相反するものではない。



「快適な立体視」

- 「視差(立体感)が強すぎて見づらい。」
 - 必要以上に視差を付けすぎると見づらくなります。
 - 視差設定を間違っていると見づらくなります。
- 「見やすいが立体感が薄い。」
 - 視差設定を間違っているても、視差を弱くすることで、見やすくはなります。

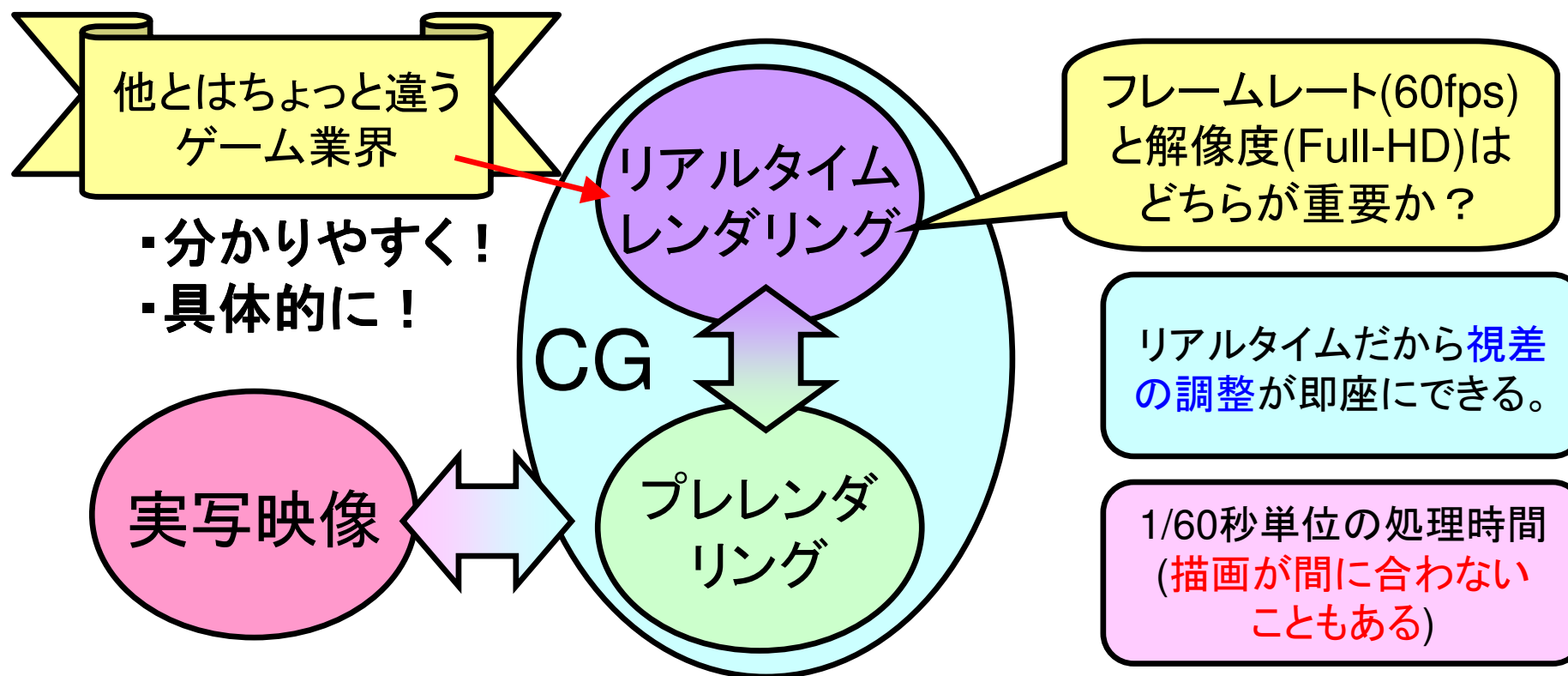


- 正しい視差設定を行うことがもっとも大切。
 - 個人差によって、視差を弱めにしたほうが見やすい場合も

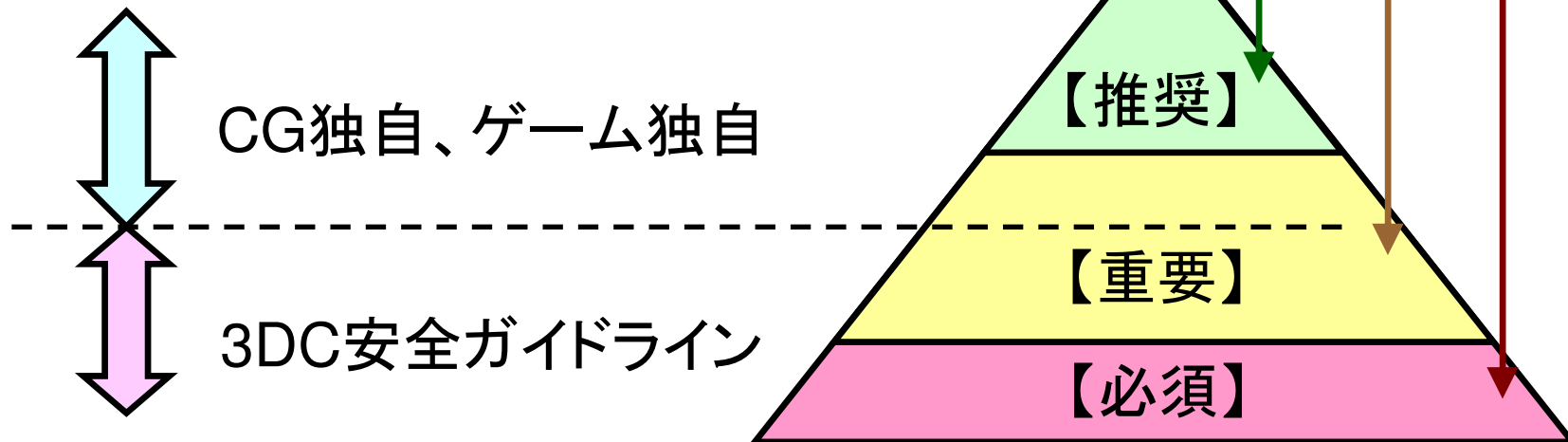
「リアルタイムゲームへ適用」

「快適な3Dを実現するためには、さらに多くのことを考慮する必要があります。
3D生体影響文献抄録集(2010)、JEITAの3次元映像に関するガイドライン試案(1999)等
を参考にして、社内又は業種別の安全ガイドライン作りにお役立てください。」

※NBGIは、JEITA主催の「3Dの生体安全性国際標準化に関する分科会」に参加しています。



- **【必須】必ず守るべきこと**
- **【重要】見やすくするために**
 - (A) 3DC安全ガイドラインを拠り所に
 - (B) 3DC安全ガイドラインには書かれていないが立体視関係者に良く知られているもの
- **【推奨】もっと見やすくするために**
 - 立体視+ゲーム独自のもの



両眼視差方式の原理に基づく『快適な立体視』を リアルタイムゲームへ適用するには



1. 【必須】必ず守るべきこと

- 間違えると明らかにおかしい

2. 【重要】見やすくするために(A)

- 3DC安全ガイドラインを拠り所に

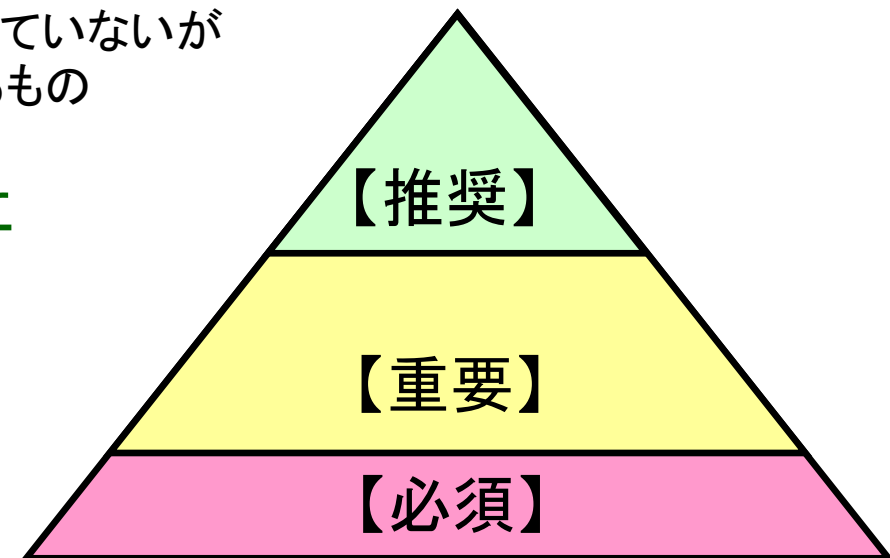
3. 【重要】見やすくするために(B)

- 3DC安全ガイドラインには書かれていないが
立体視関係者に良く知られているもの

4. 【推奨】より見やすくするために

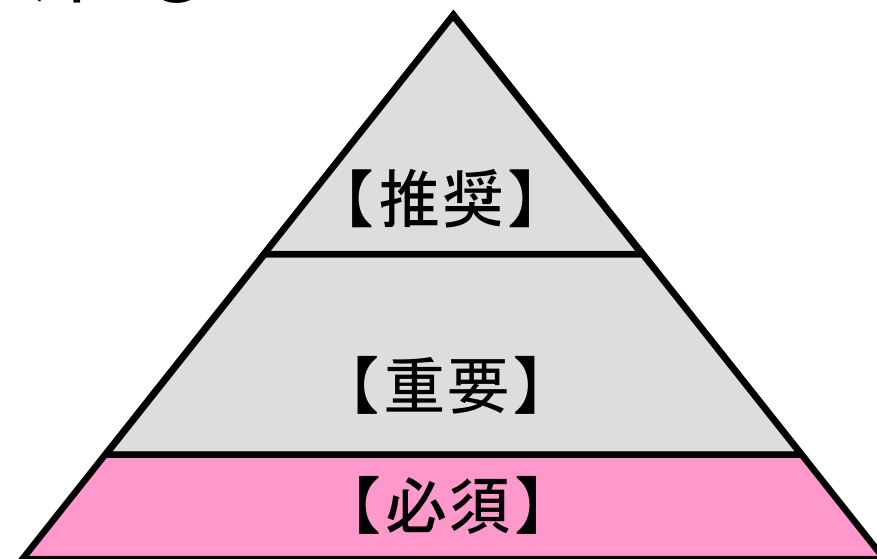
- 立体視+ゲーム独自のもの

5. 補足とまとめ



1. 【必須】必ず守るべきこと

- (0) 左画像・右画像を間違えない
- (1) 垂直視差の排除
- (2) 両眼間隔に合わせたカメラ設定
- (3) 画面サイズを考慮に入れること



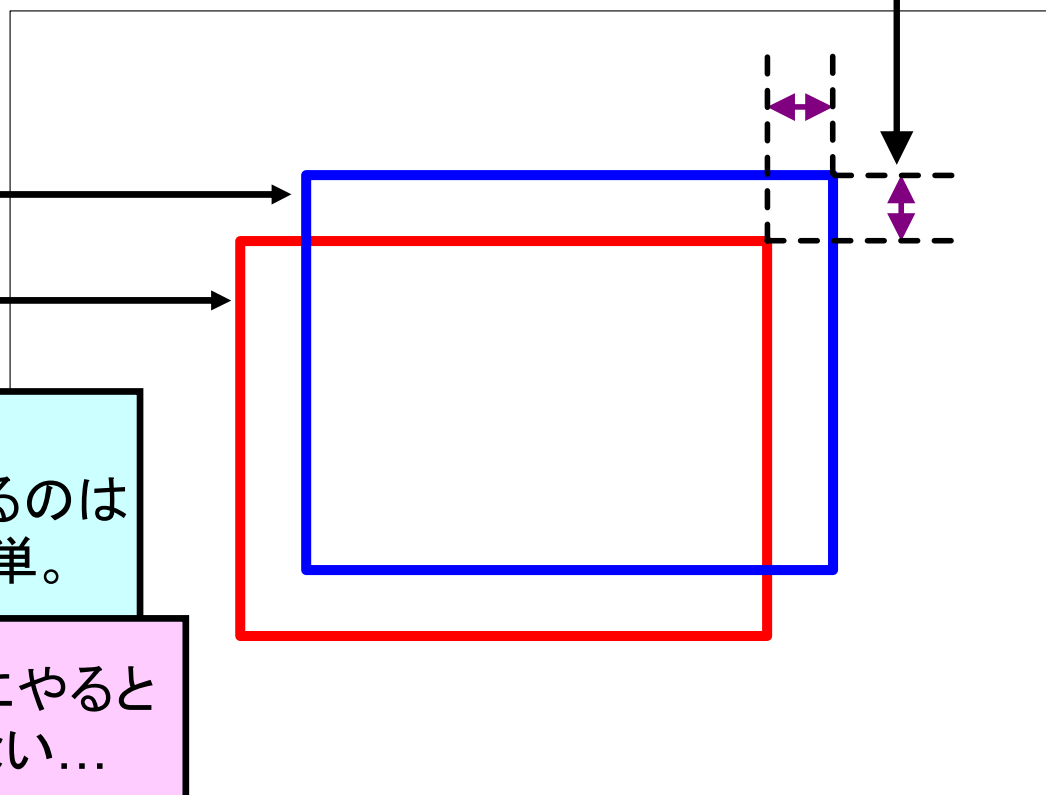
(1) 垂直視差の排除

- 垂直ずれをなくす。

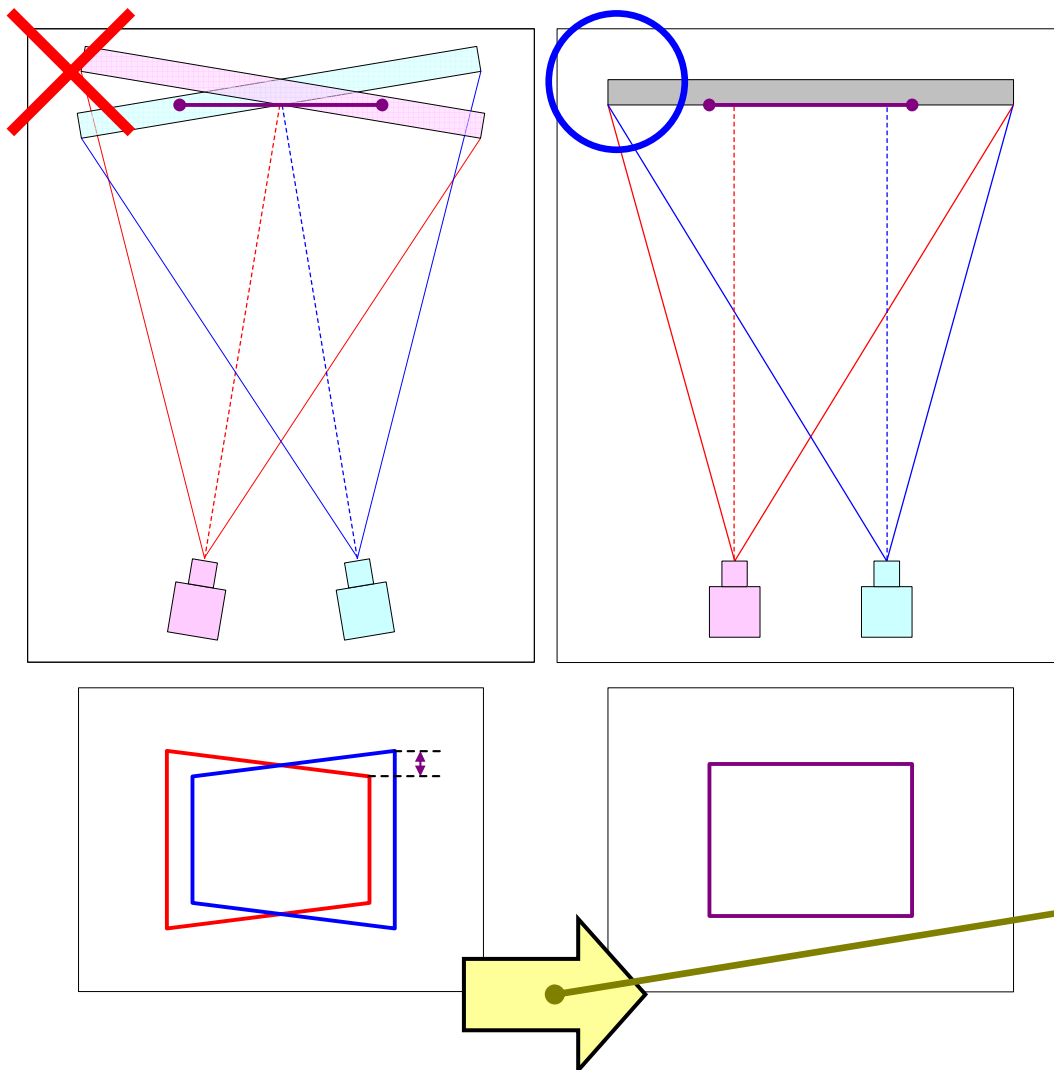
右眼映像
左眼映像

実写だと2台のカメラを合わせるのは
非常に大変だが、CGだと簡単。

しかし、実際にやると
うまく行かない...



スキューした視錐台



「通常の視錐台」の問題点

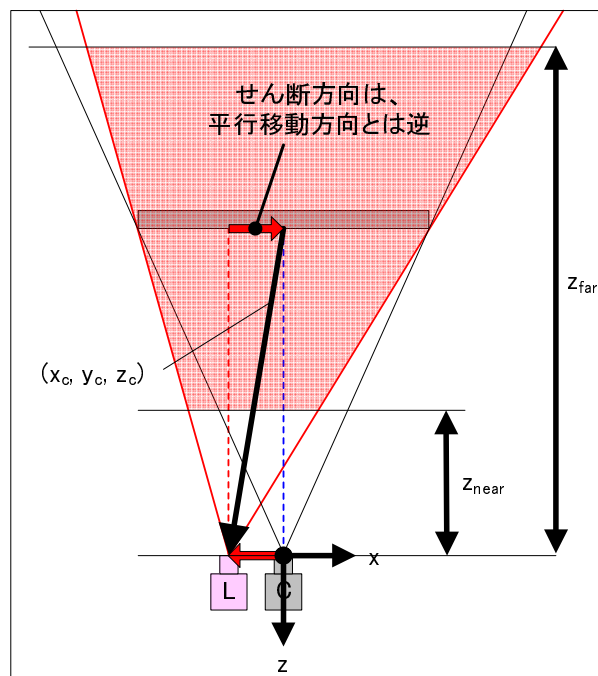
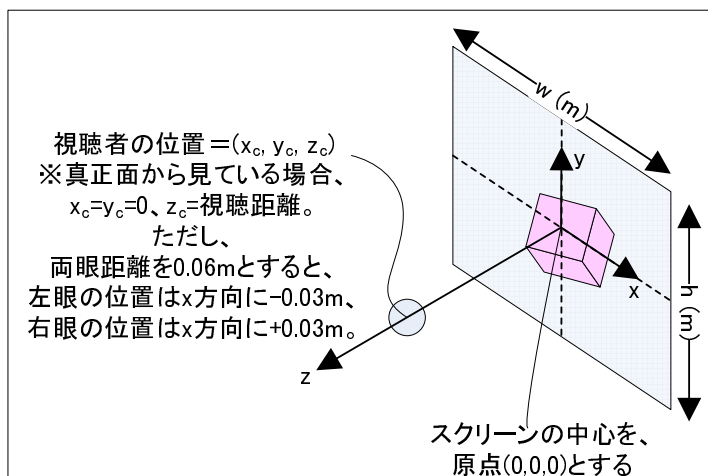
- キーストーン(台形)歪み
→ 垂直視差
- 遠方が発散

「スキューした視錐台」は、
実写カメラでは物理的な制約
や価格の影響を受けるが、
CGではノーコスト。

実写では、普通に撮影した
ものを画像処理で修正する
場合もある。

→ 「遠山式立体表示法」(PAT.)

スキューした視錐台



$$\begin{pmatrix} x & y & z & 1 \end{pmatrix} \begin{pmatrix} 2z_c/w & 0 & 0 & 0 \\ 0 & 2z_c/h & 0 & 0 \\ -2x_c/w & 0 & \alpha & -1 \\ 0 & 0 & \beta & 0 \end{pmatrix}$$

この部分にスキュー(せん断)成分が入る。
それ以外は通常と同じ。

$$\begin{pmatrix} 2z_c/w & 0 & -2x_c/w & 0 \\ 0 & 2z_c/h & 0 & 0 \\ 0 & 0 & \alpha & \beta \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

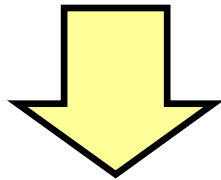
※ α 、 β は z_{near} と z_{far} で決まるが、処理系によって異なる。
(これも通常と同じ)

	OpenGL系 : zが-1~+1	DirectX系 : zが0~+1
α	$-(z_{far} + z_{near}) / (z_{far} - z_{near})$	$-z_{far} / (z_{far} - z_{near})$
β	$-2 \times z_{far} \times z_{near} / (z_{far} - z_{near})$	$-z_{far} \times z_{near} / (z_{far} - z_{near})$

- プログラムの話

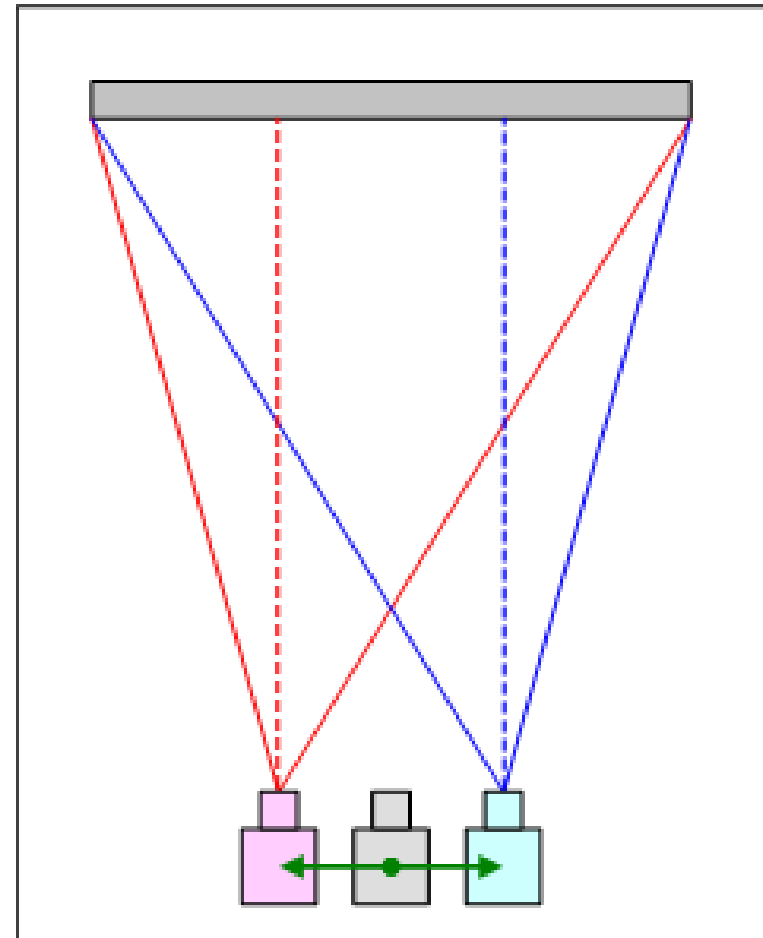
※両眼間隔を60mmとします。

```
Translate(±30, 0, 0); //mm?  
Translate(±0.03, 0, 0); //m?
```

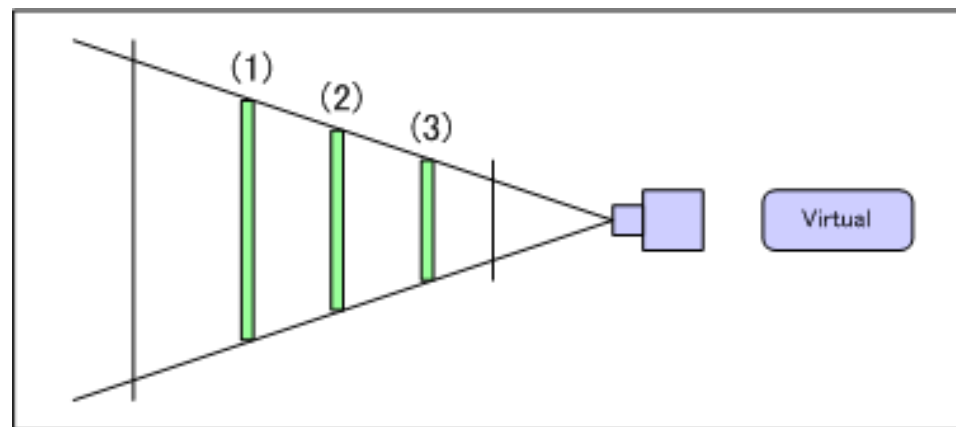


- 「スケール」の設定

- 現実空間の長さを
仮想空間の長さに変換する倍率

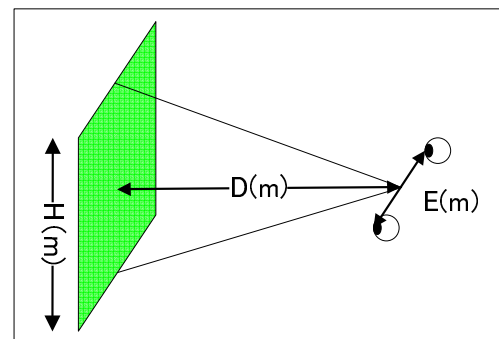


- 平面視では、スクリーンはどこにおいても同じだが、立体視では違う。
- 「スケール」は、「スクリーンをどこに置くか」
- 両眼間隔、視聴距離、
スクリーンサイズは、1つの「系」
 - 巨大化、縮小化するとき注意



Translate($\pm 0.03 \times \text{scale}$, 0, 0);

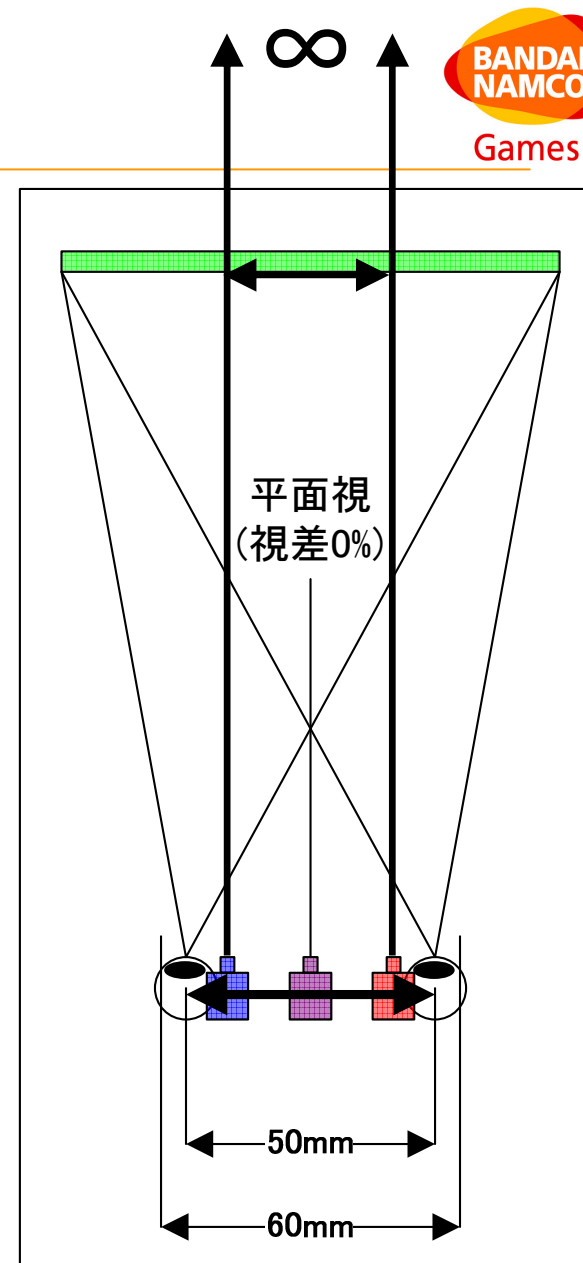
現実の1mは、
仮想空間内の長さでいくら？



(2) 両眼間隔に 合わせたカメラ設定

※まず、「大画面」を前提に話します。

- 両眼間隔=「瞳孔間隔」、「両眼幅」
 - 大人60mm, 子供50mm
 - 「個人差」がある。
 - プレイヤーが自分で調整できない場合には、子供に合わせる。
- プレイヤーが自分で調整できる場合
 - 「視差を上げるほどすばらしいに違いない」とプレイヤーに思い込ませない

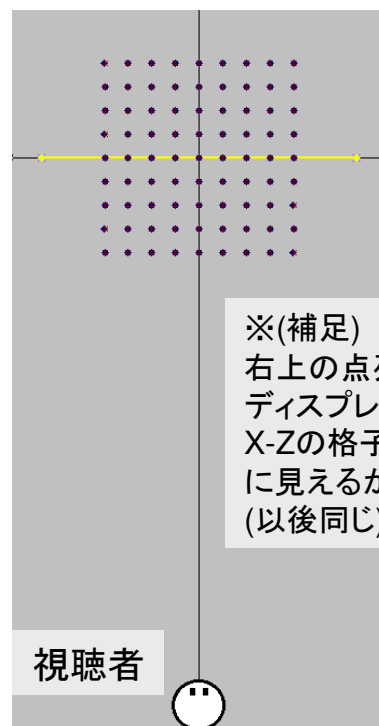


過視差による後方発散

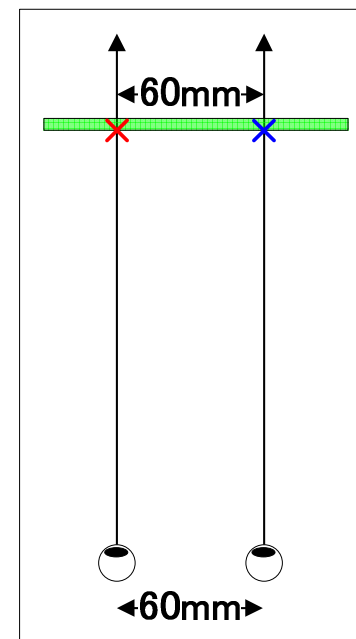
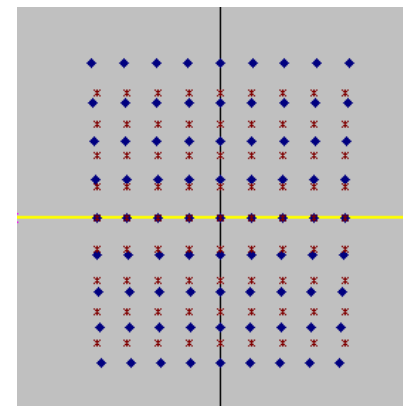
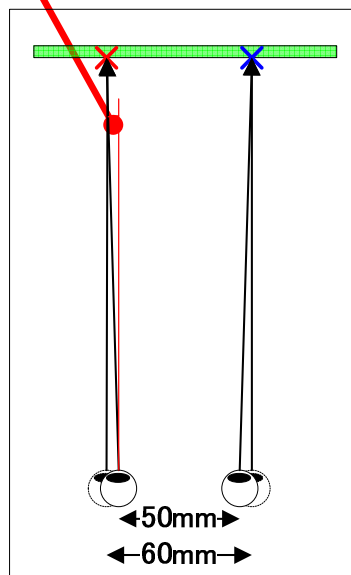
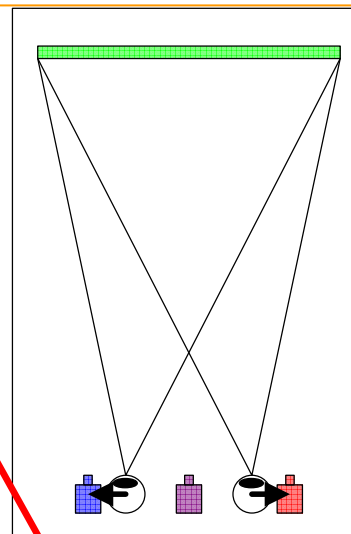
• 大人設定のものを子供が見るとどう見えるか？

- 無限遠で「開散方向の視差」
- 疲労の原因に。
- 子供が遊ぶゲームでは、両眼間隔は50mmまでに

- コンテンツ制作者も「個人差」の例外ではない。

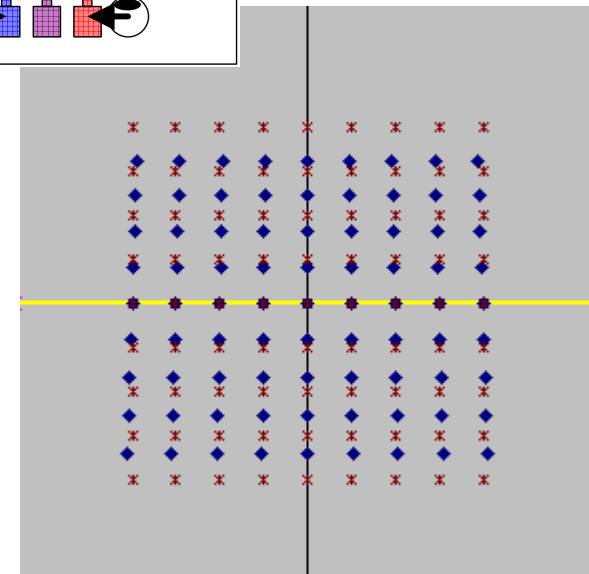
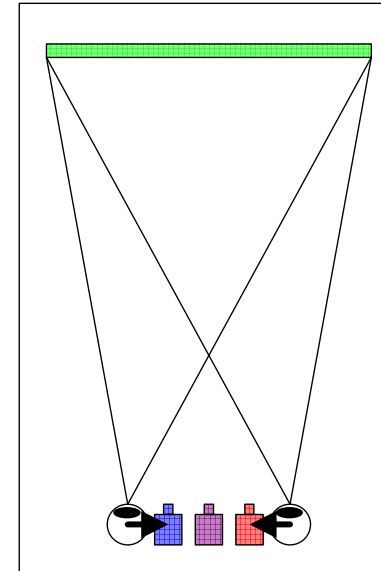


※(補足)
右上の点列図は、立体視ディスプレイに表示されたX-Zの格子点が、どのように見えるかを示しています。(以後同じ)



視差が弱くなる分には問題ない？

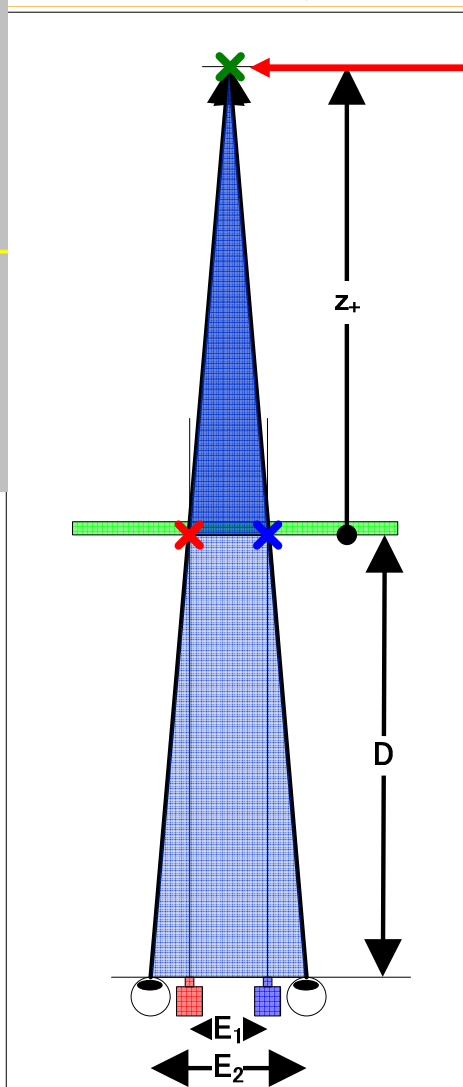
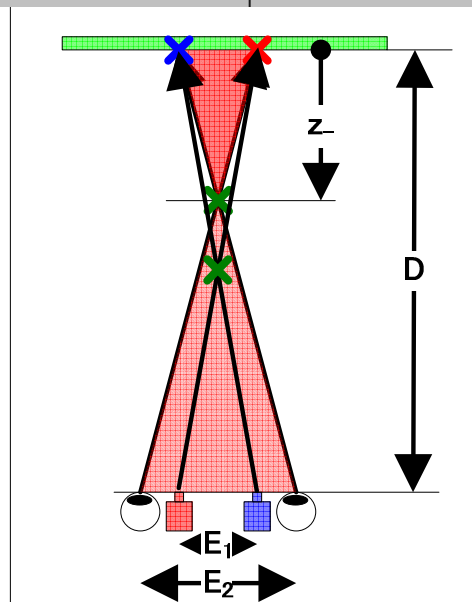
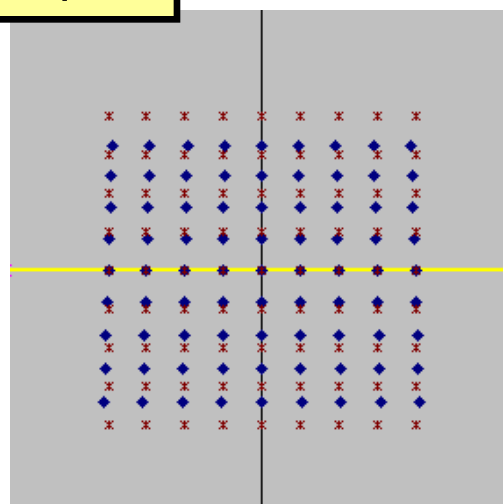
- 「視差が弱くなる分には問題ない(はず)」の大前提
 - 視差0(平面視)はOK,
自分と同じ両眼間隔設定はOK.
→両者の補間はOK(のはず)
 - 「距離感の把握」が重要な場合には、
空間感覚に違和感が出ることも。
- 立体感が少なくて迫力不足にならないか？



1-(2)

GL-p5

立体感が少なくて 迫力不足にならないか？



無限遠が有限距離に

$$E_2 : D + z_+ = E_1 : z_+$$

$$z_+ = \frac{E_1 D}{E_2 - E_1} = 5D$$

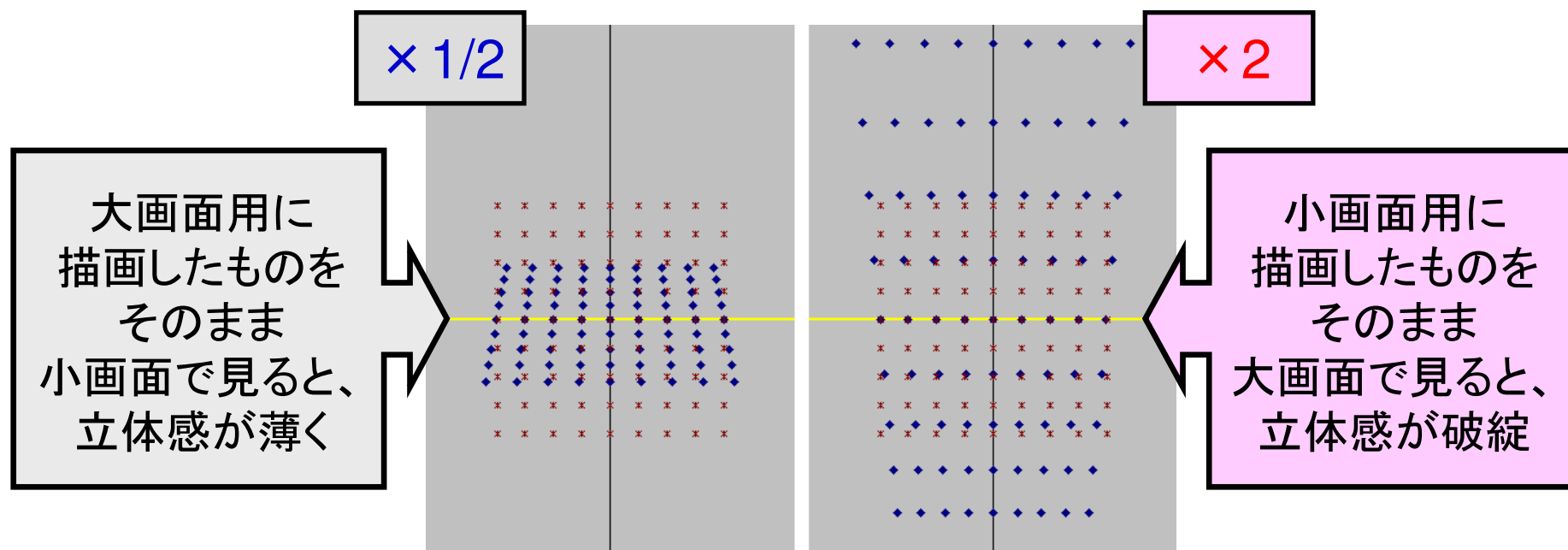
$$E_1 = 50\text{mm}, E_2 = 60\text{mm}$$

世界全体が縮まって見える
だが、カメラ間隔を元通りに
広げるわけには行かない...

縮まった立体感は、
カメラ間隔を広げる以外の方法
でカバーする。

(3) スクリーンサイズを意識すること

- スクリーンサイズは、平面視画像ではあまり意識されていないが、立体視では...

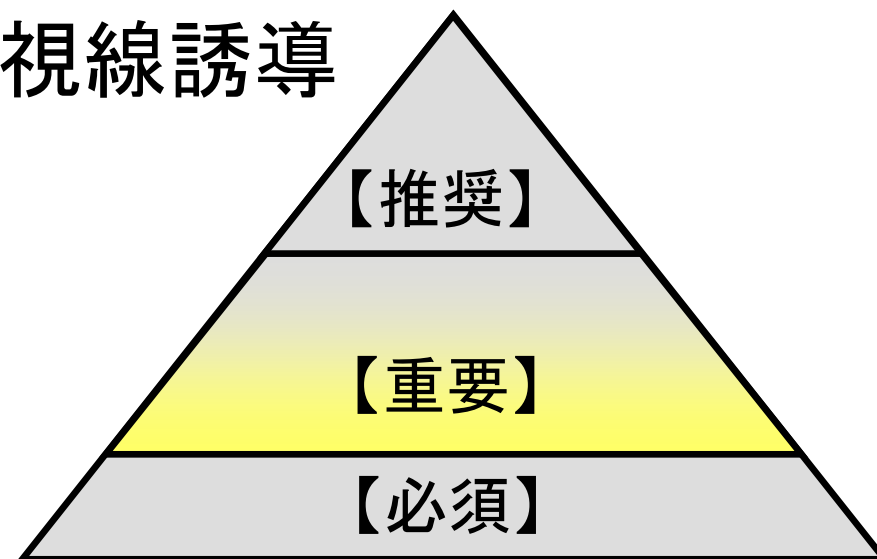


- サイズに合わせて正しく描画できるような設定項目を。
 - 「画面サイズの設定」は正しくされていますか？
- 画面サイズの違う移植、ショー会場でのデモの時には注意。

2. 【重要】見やすくすること(A)

※「3DC安全ガイドライン」の記述を元に。

- (1) 快適・限界範囲
- (2) 「視差角」を「使える量」に変換
- (3) 大画面と小画面の違い
- (4) 奥行き範囲の設定と視線誘導



まず、「視差角」の定義から

- 輻輳角＝両眼を寄せる角度。
- 視差角

－ 画面での輻輳角を α とする。

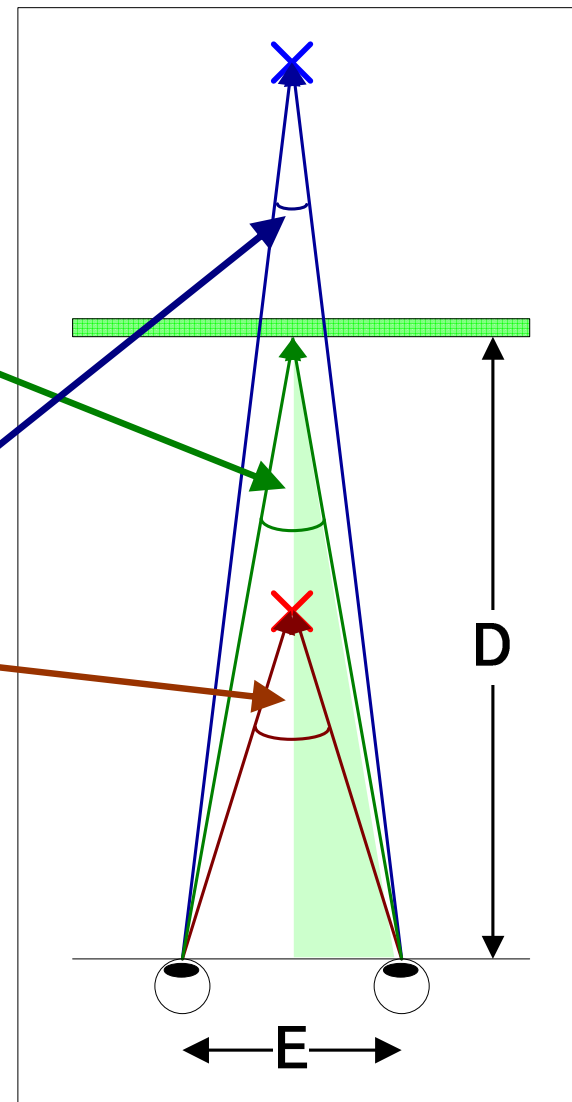
$$\tan \frac{\alpha}{2} = \frac{E/2}{D} \rightarrow \alpha = 2 \arctan\left(\frac{E/2}{D}\right)$$

－ 注目点の輻輳角を β とする。

－ 視差角 $= \alpha - \beta$

- 画面(視差0面)で視差角 $= 0$
- 飛び出し方向: $-$
- 奥行き方向: $+$

※「3DCガイドライン」の記述とは、 α 、 β の定義が異なります。ご注意ください。



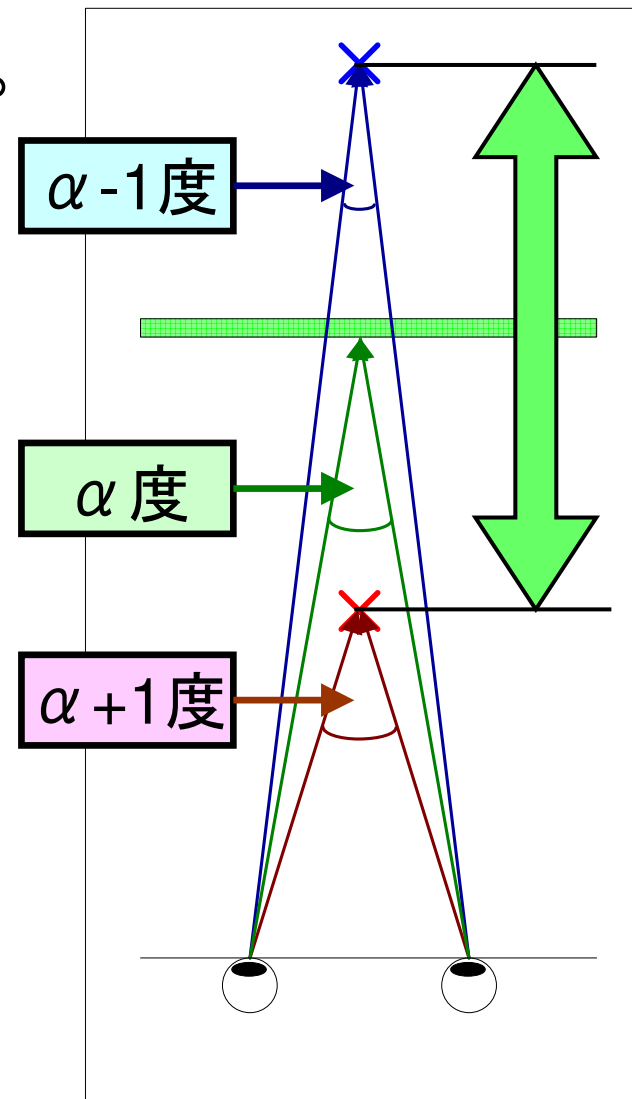
(1) 快適・限界範囲

- 視差角0の位置が最も見やすい。
(「輻輳・調節矛盾」がないから)

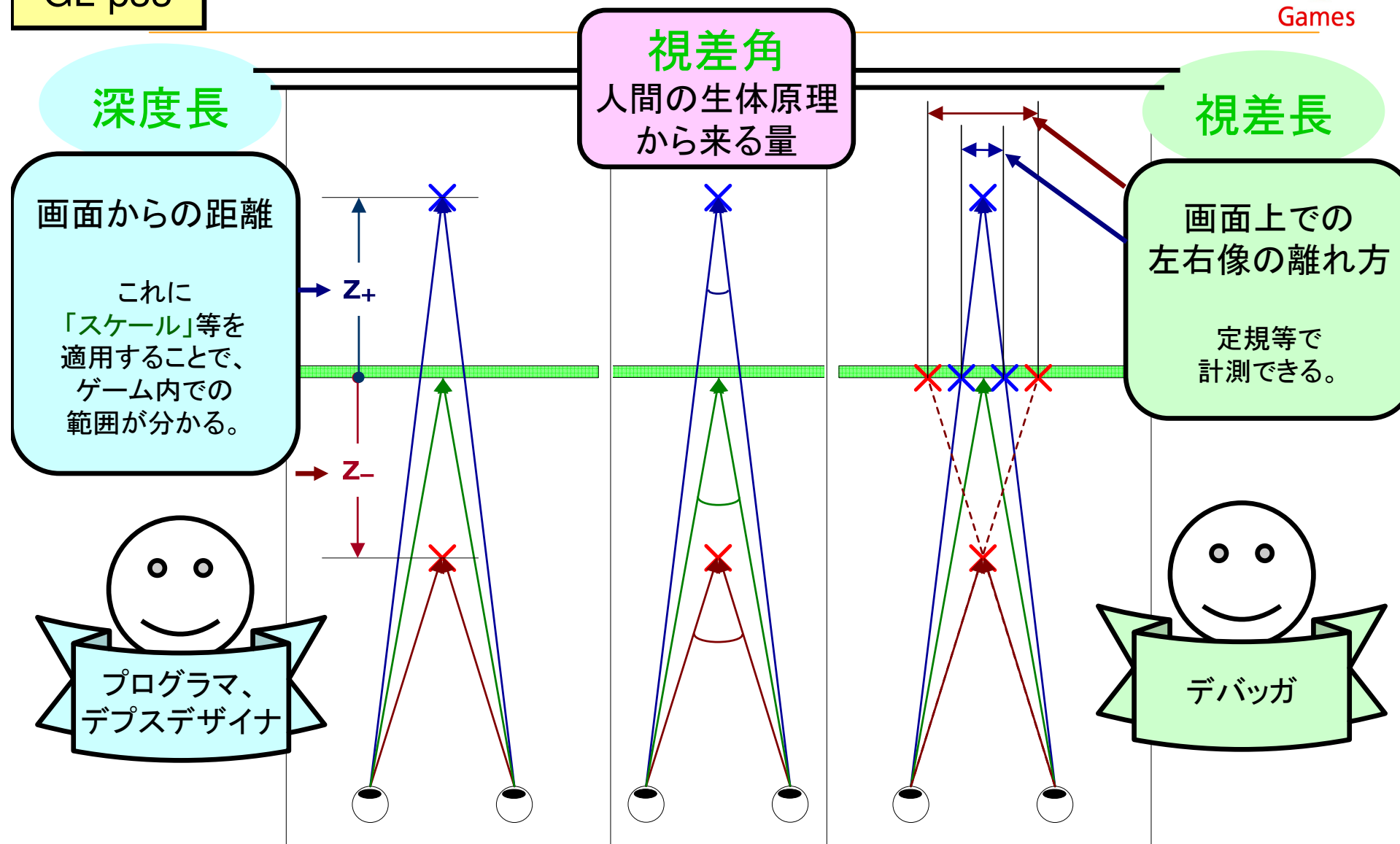
- 快適・限界範囲

(輻輳・調節矛盾の許容範囲)

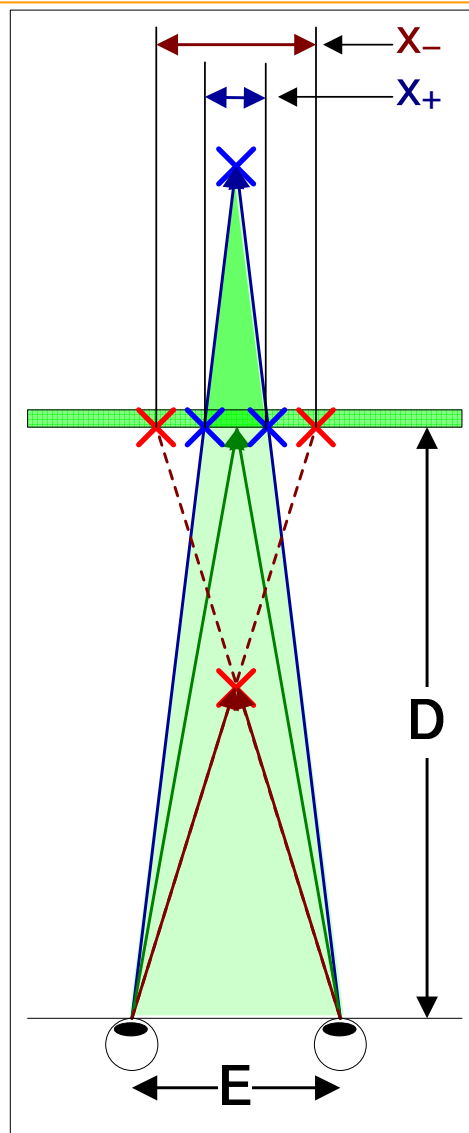
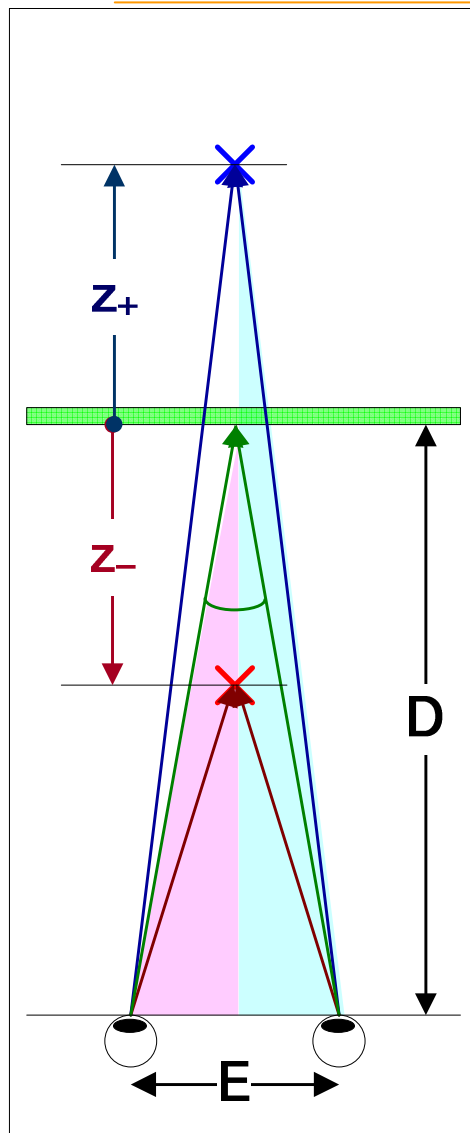
- 視差角が ± 1 度以内
「快適範囲」
- 視差角が ± 2 度程度以内
「融合限界範囲」の下限側
 - 個人差が大きい。
 - 訓練によって拡大する。
 - 映像製作者は鍛えられているから、大きめでも問題ないように感じてしまう。



(2) 「視差角」を「使える量」に変換



「視差角」⇔「深度長」「視差長」換算



$$\tan \frac{\alpha}{2} = \frac{E/2}{D}$$

$$\tan \frac{\alpha - 1}{2} = \frac{E/2}{D + z_+}$$

$$E : D + z_+ = x_+ : z_+$$

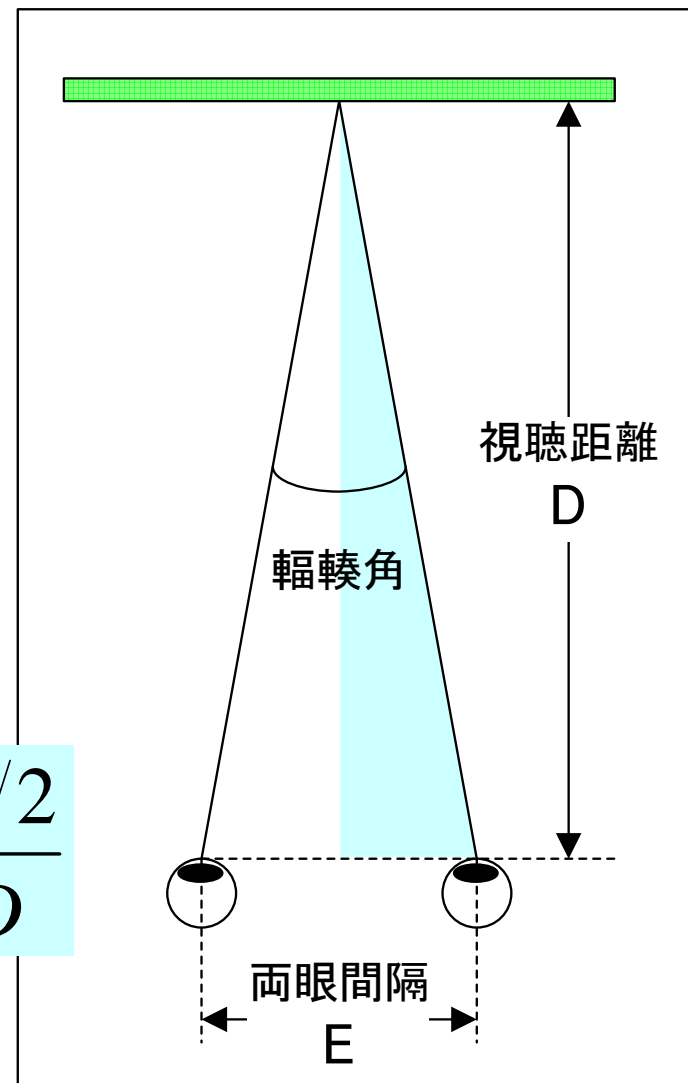
※飛び出し側も同様

→ツール化

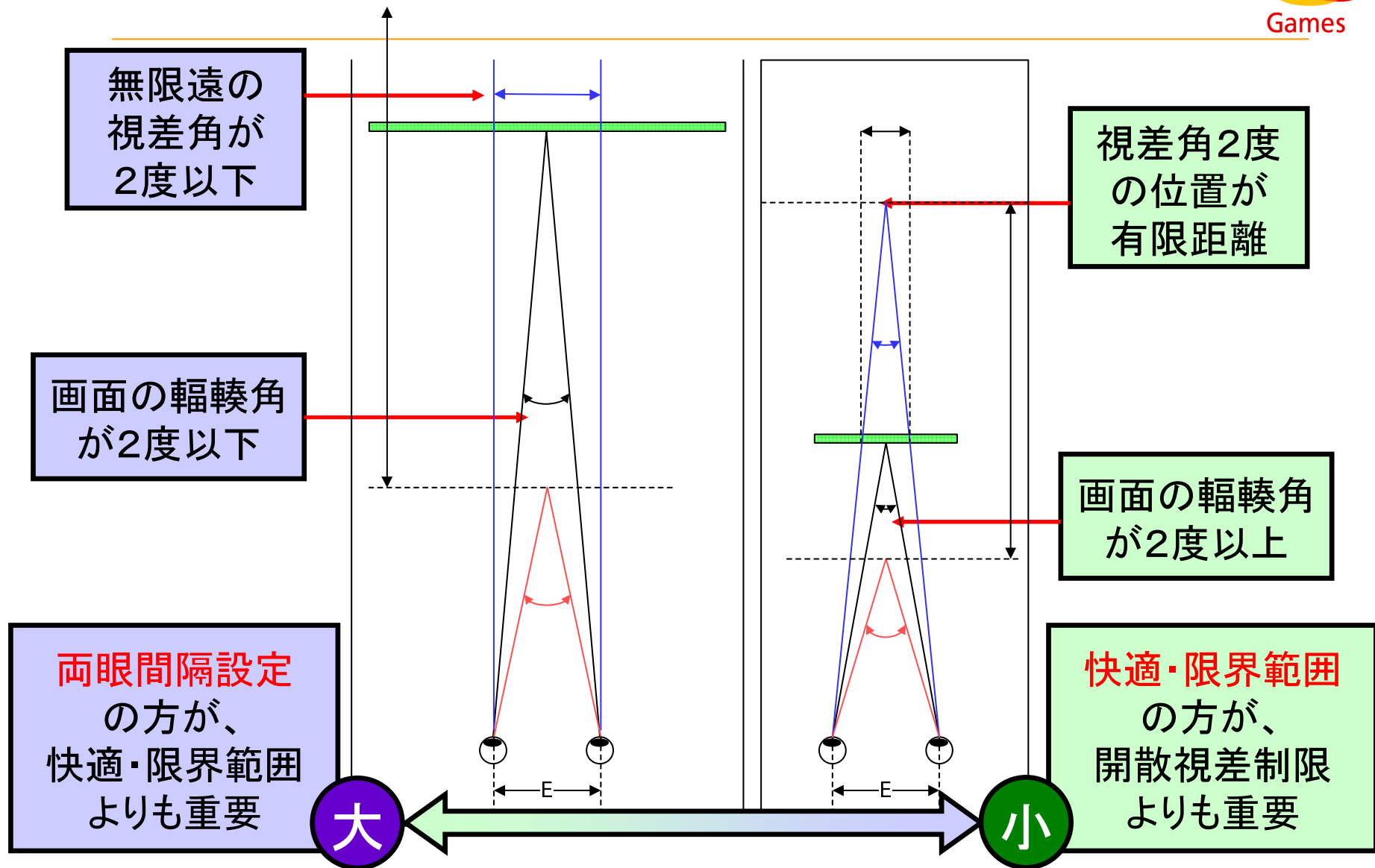
(3) 大画面と小画面の違い

- 視差角の範囲は、視聴距離によって変わる。
 - 両眼間隔 $E=60\text{mm}$ の場合、画面の輻輳角 α が「2度」になる視聴距離は1.7m、「1度」になる視聴距離は3.4m。
 - 大画面では、快適・限界範囲内で無限遠を表現できる。

$$\tan \frac{\alpha}{2} = \frac{E/2}{D}$$

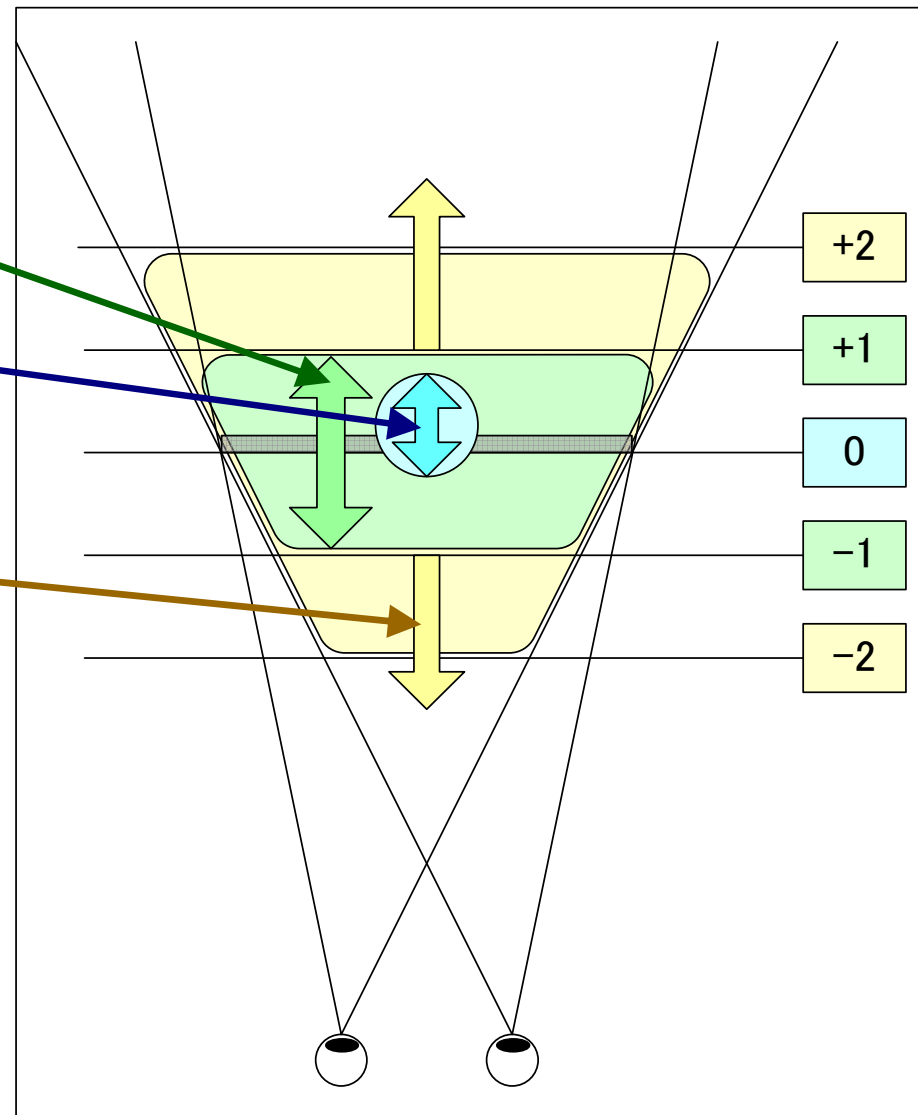


大画面と小画面での、留意点の違い



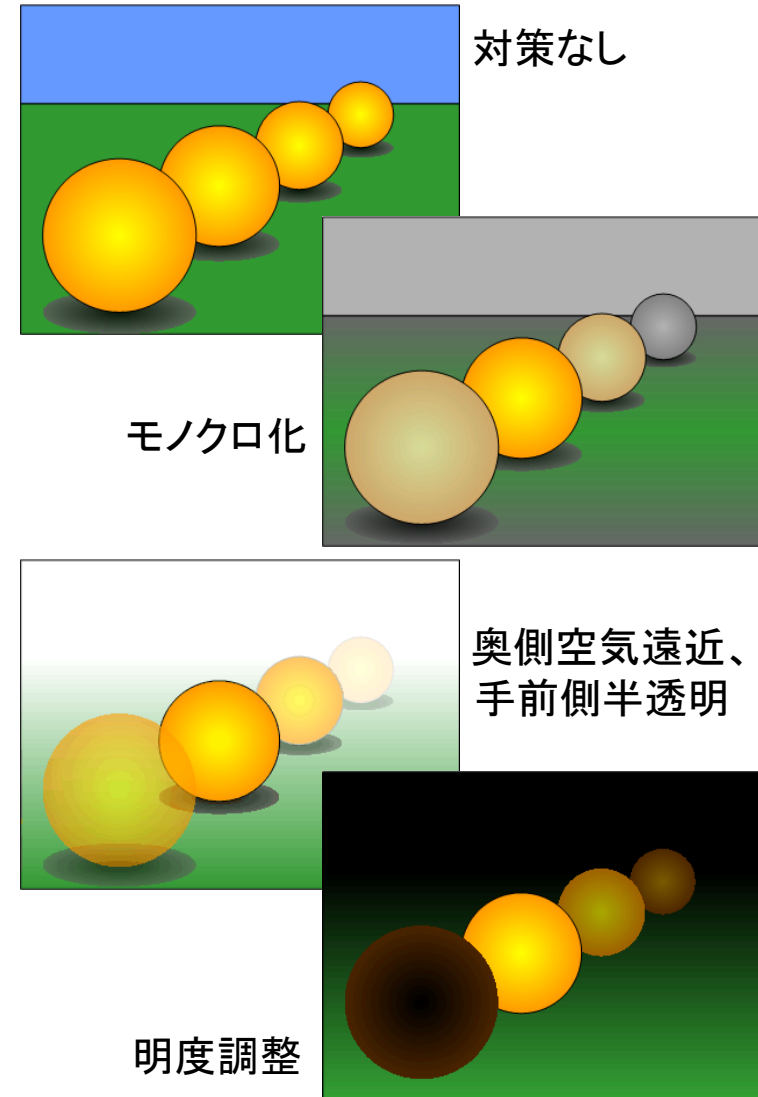
(4) 奥行き範囲の設定と視線誘導

- 注目が行きがちなもの
 - 主人公等は、
±1度の範囲に置き、
オブジェクト内の視差を
1度以内に収める。
- それ以外のもの
 - なるべく**±2度以内**に
 - 範囲を外れるにつれ、
注目が行かないような工夫を
 - 手前側は**注目が行きがち**
なので特に注意
- 製作側が意図したところに注目し
てくれるとは限らない...



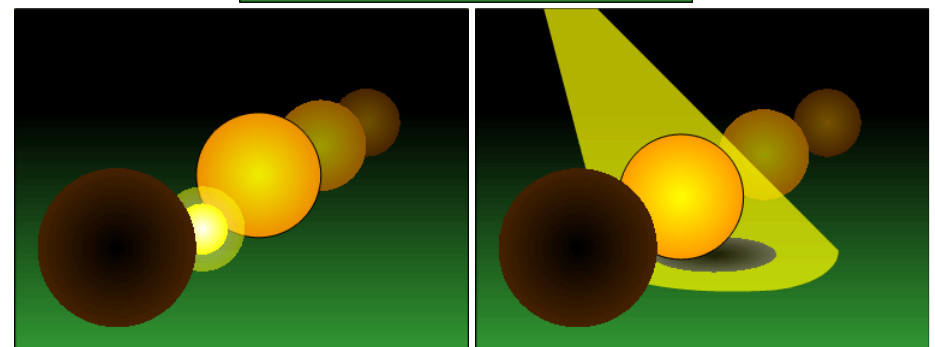
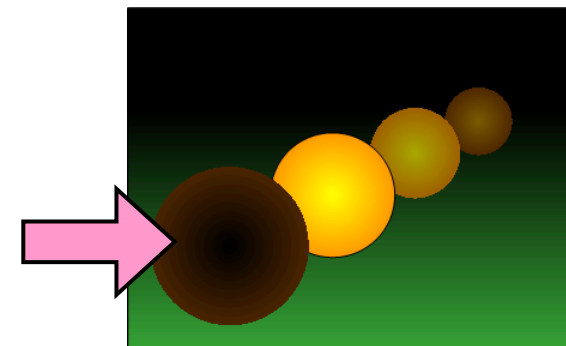
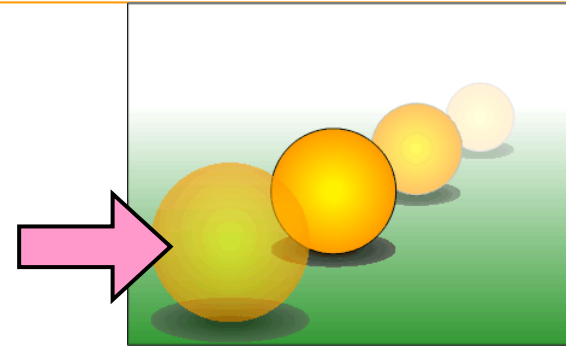
(4) 奥行き範囲の設定と視線誘導

- 鮮鋭度 (例: 被写界深度)
 - 実写では理にかなっている。
 - CGでは、やや処理が重い。
 - 立体視の醍醐味が失われる場合も。
- 彩度 (例: モノクロに近づける)
 - CGの方が得意な処理。
 - 「奥側空気遠近、手前側半透明」
 - 「半透明」は、やや処理が重い
- 明度 (暗くする)
- その他の目立たなくする工夫
 - すぐ通り過ぎる、消えてなくなる、「パーティクル」である
- 立体感がアップし、クロストーク(2重映り)を減らせる効果もある。



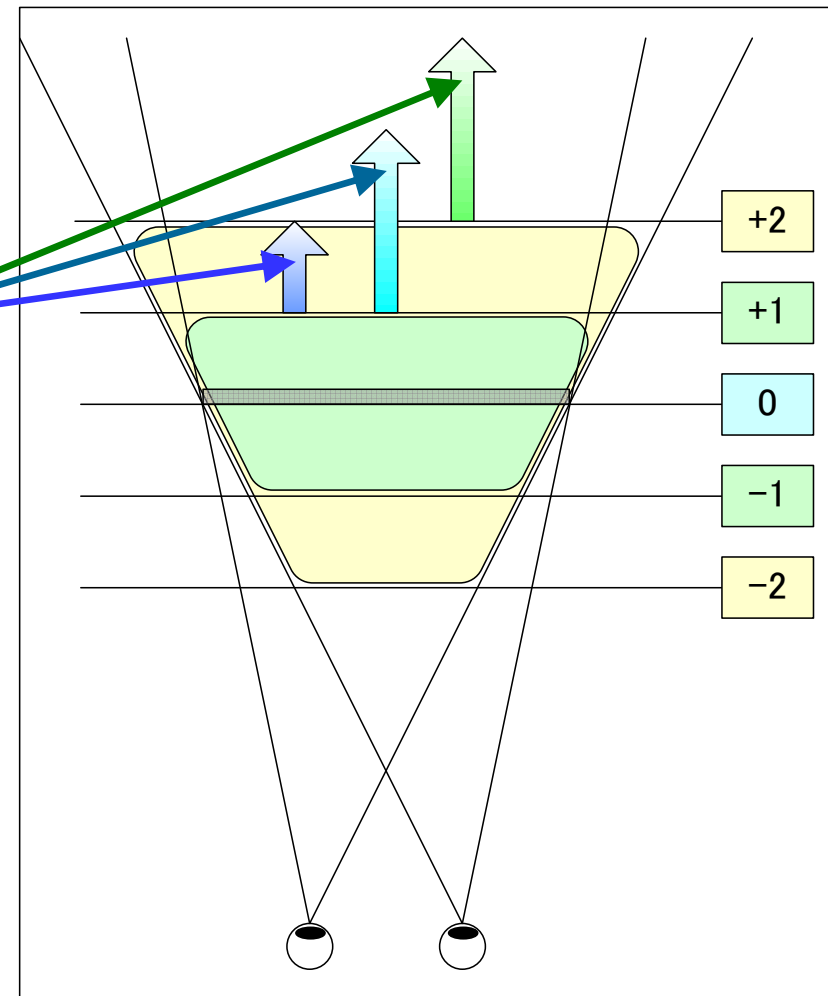
「不自然じゃない？」

- 「半透明」は不自然？
 - 使い方次第
 - 分かりやすさ優先
- 「手前側が黒い」のは不自然？
 - ライトワークによる
- 「不自然じゃない方法」
「さりげない方法」を選んで
使い分ける。
 - 「映画に学べ。」



奥行き範囲設定の実際

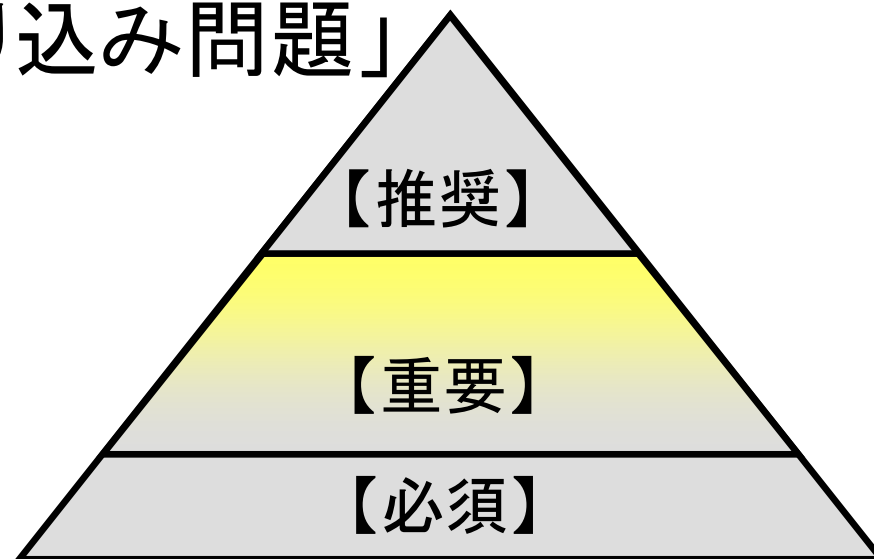
- 「快適・限界範囲」の適用方法は様々。
- あまり厳しく規定せず、現場に任せる。
 - 例えば空気遠近で消す場合...
- ただし、「快適・限界範囲」の意味を理解し、意識すること。
 - 見やすさに影響する。
 - 「客観的に見る」基準となる。
 - ユーザーの評価を、効果的にフィードバックできる。



3. 【重要】見やすくすること(B)

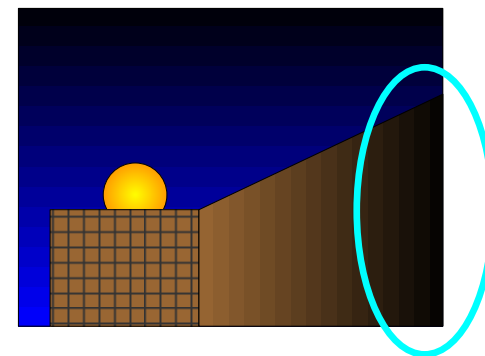
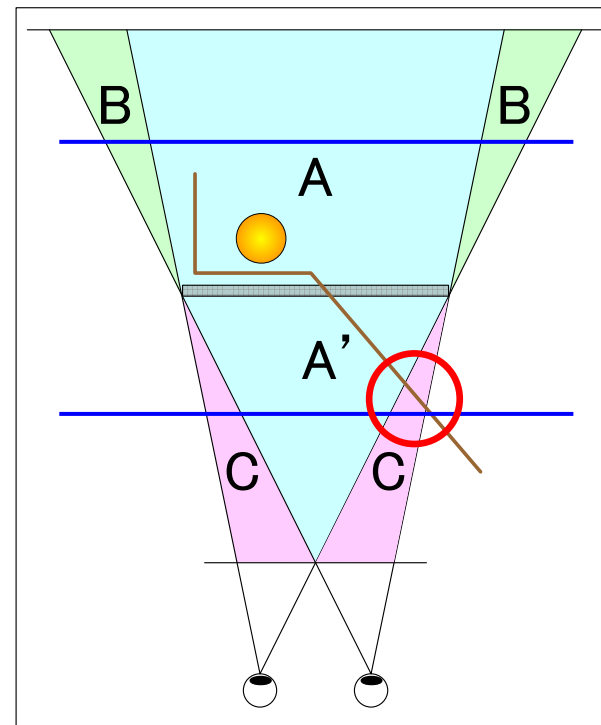
※「3DC安全ガイドライン」に書かれていないが、有名なもの。

- (1) フレームバイオレーション対策
- (2) 奥行き方向の移動速度
- (3) 「奥行き乖離」と「めり込み問題」



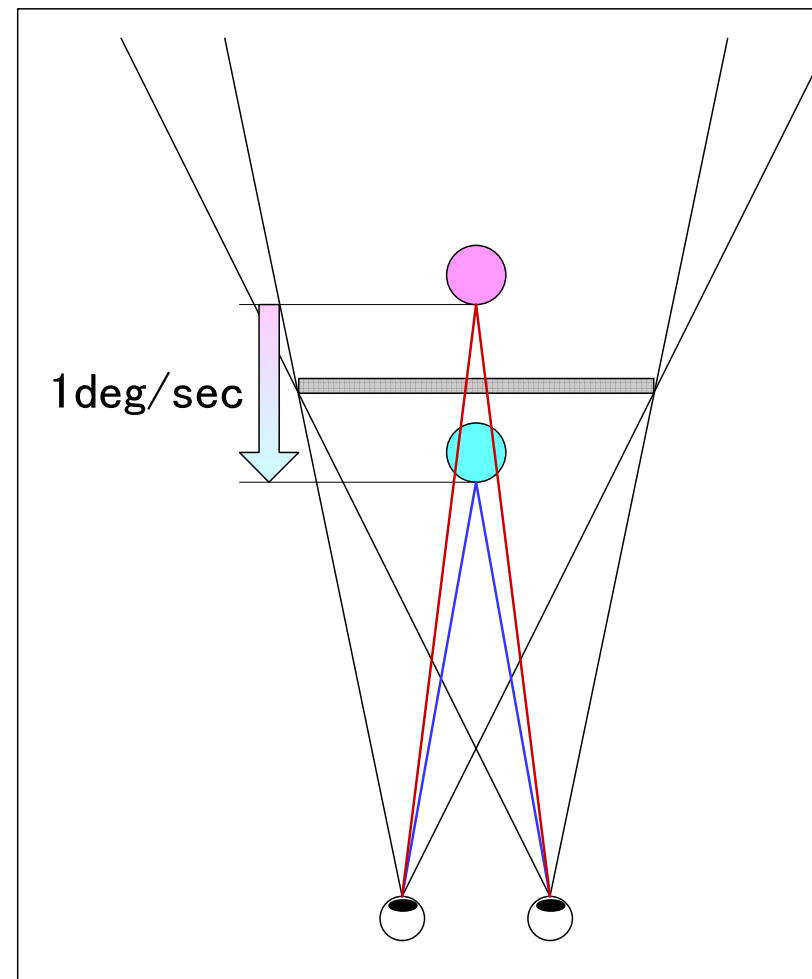
(1) フレームバイオレーション対策

- フレームバイオレーション
(ウィンドウ違反)
- 飛び出させてはいけないのか？
 - A' は、「飛び出し」が許された貴重な範囲
 - 「快適・限界範囲」を超える場合は工夫を
- フレームバイオレーション領域
(BとC, 特にC)
 - 注目する物体が、この領域に来ないようにする。
 - 目立たなくする。
 - これらの対策が取れないのなら、
「すべてを奥行き側に」

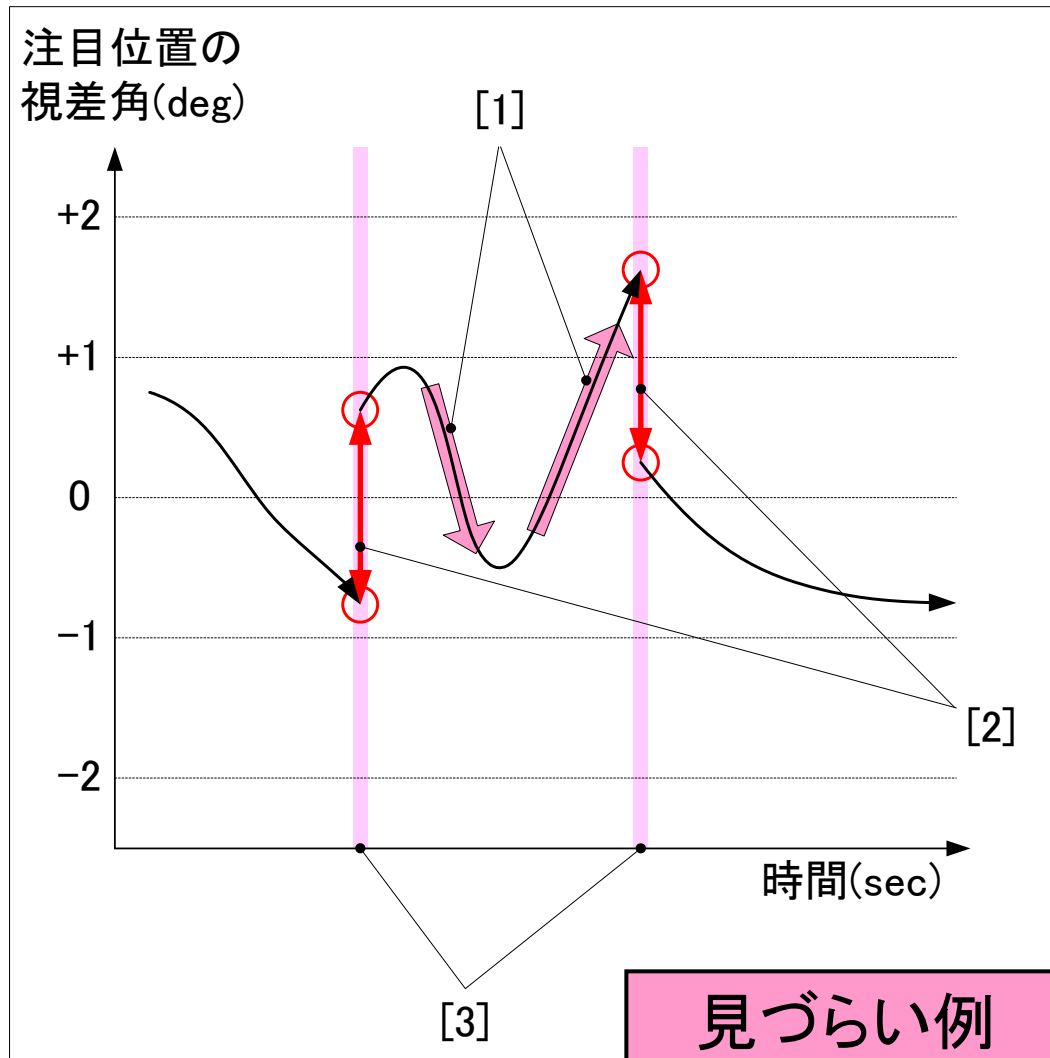


(2) 奥行き方向の移動速度

- 「1度/秒」なら輻輳が追従できる、
という調査結果がある。
 - フレームレートや視聴距離にも
拠るので、あくまで目安として
 - 注目して欲しいものは、
ゆっくり移動。
 - 注目して欲しくないものは、
素早く移動。

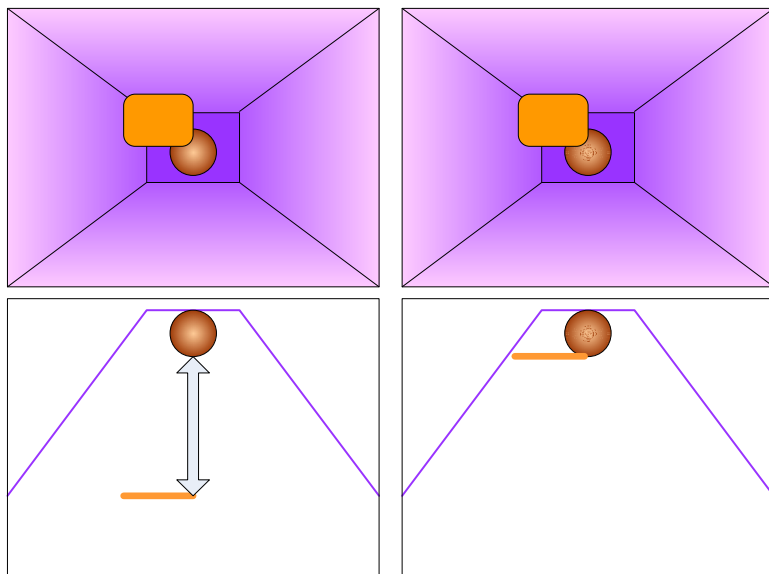


(2) 奥行き方向の移動速度など

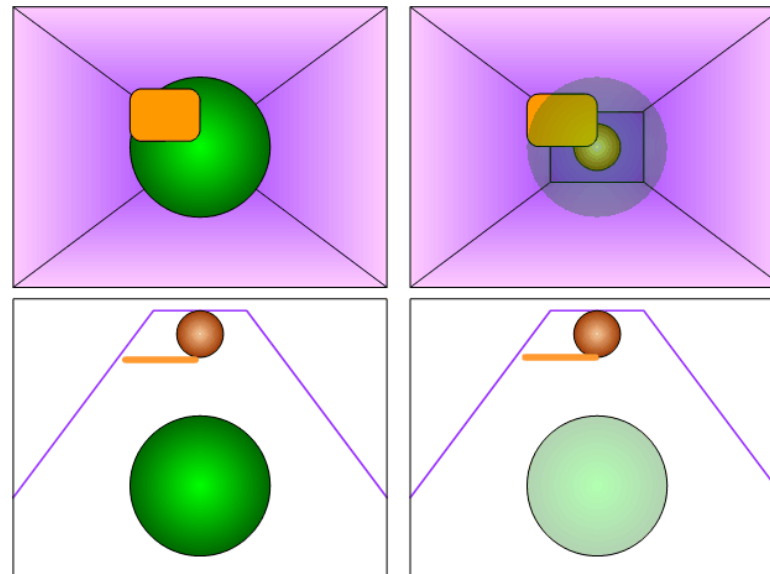


- [1] 注目点の移動は
ゆっくりと
- [2] 「奥行き飛び」の解消
- [3] あるいは、
視線が追従できるように
クロスフェード化

(3)「奥行き乖離」と「めり込み問題」

**奥行き乖離**

- 同時に見る必要がある2箇所奥行きが、分かれていて見づらいこと。
- 映画の字幕などによく見られた。
- 対策：奥行き差をなるべく少なくする。

**奥行き・隠蔽矛盾（めり込み問題）**

- 描画優先度と奥行き関係が逆転しているため、**不自然なめり込み方**に見える問題。
- HUD類と飛び出しオブジェクトでよく起こる。
- 対策例：手前側を半透明にする

(3)「奥行き乖離」と「めり込み問題」 を解決するには

- 両者をまとめて解決するのは至難の業。

- 回避例

- めり込み問題を回避するために、全てのHUD類を視差0面に置き、全ての3Dオブジェクトを奥側に配置する。

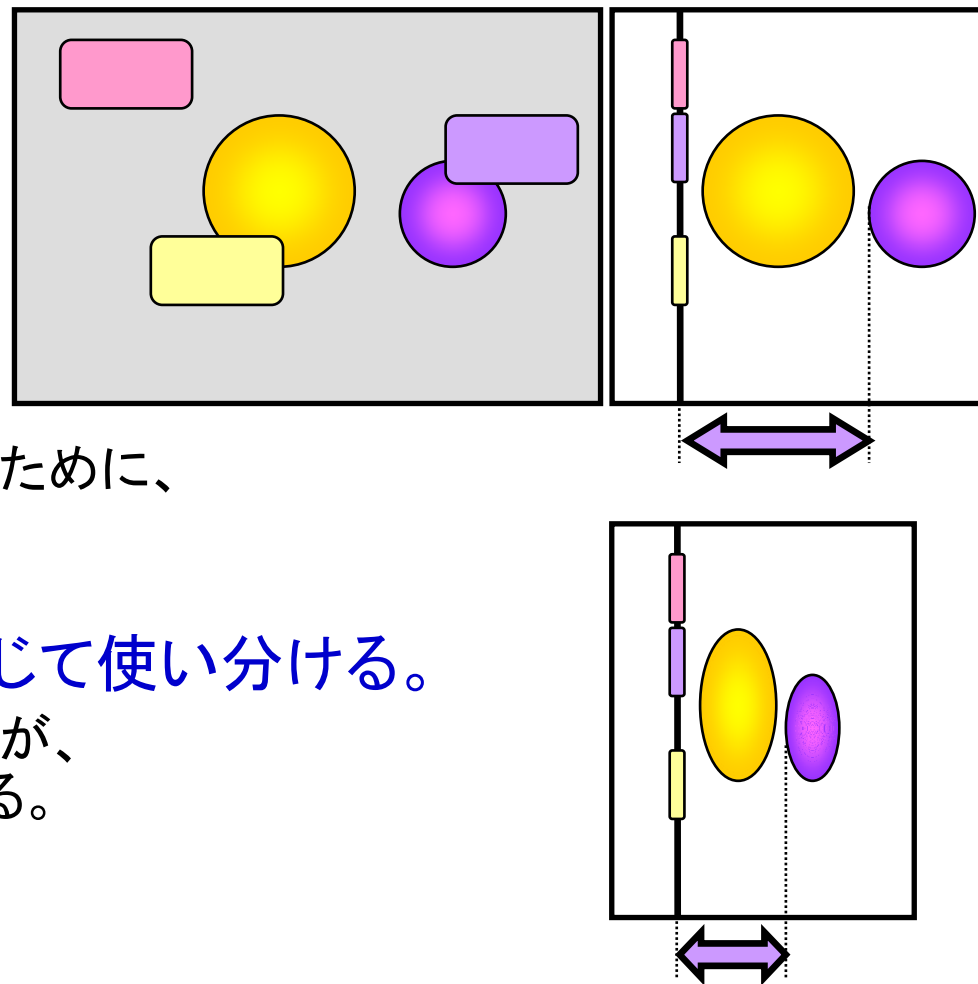
...飛び出し表現が難しくなる。

- さらに、奥行き乖離を回避するために、視差を少なくする。

...立体感が減る。

- いくつかの対策を、場面に応じて使い分ける。

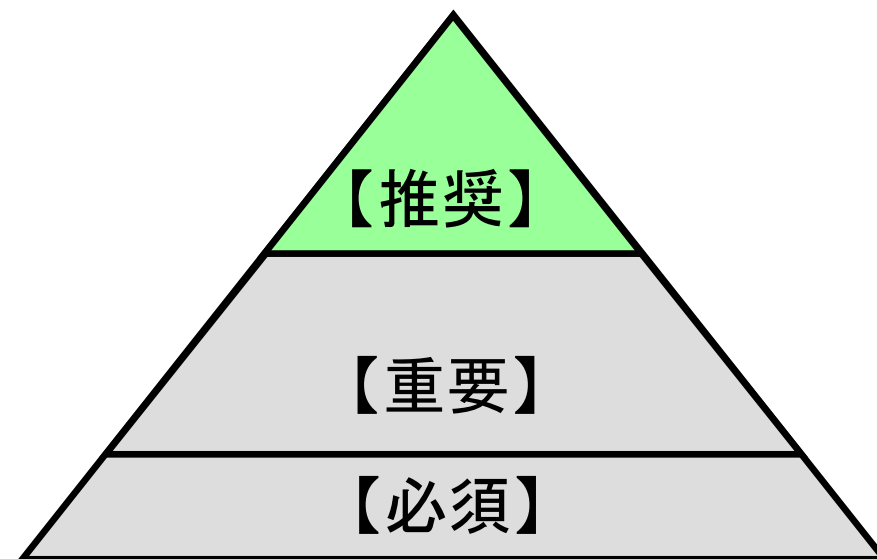
- 上記の方法は消極的ではあるが、場合によっては最善の策になる。



4.【推奨】より見やすくするために

※ゲーム寄りの内容

- (1) 細かいところが気になる...
- (2) 見落としがちなこと
- (3) 神経質になり過ぎない

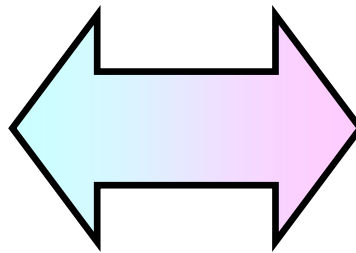


(1) 細かいところが気になる...

- 「動いて分かるものは、止まっても分かる」

平面視で

平面視で運動視差
が出たときに
美しく見える効果は、
立体視では動かさ
なくても美しい。



立体視では

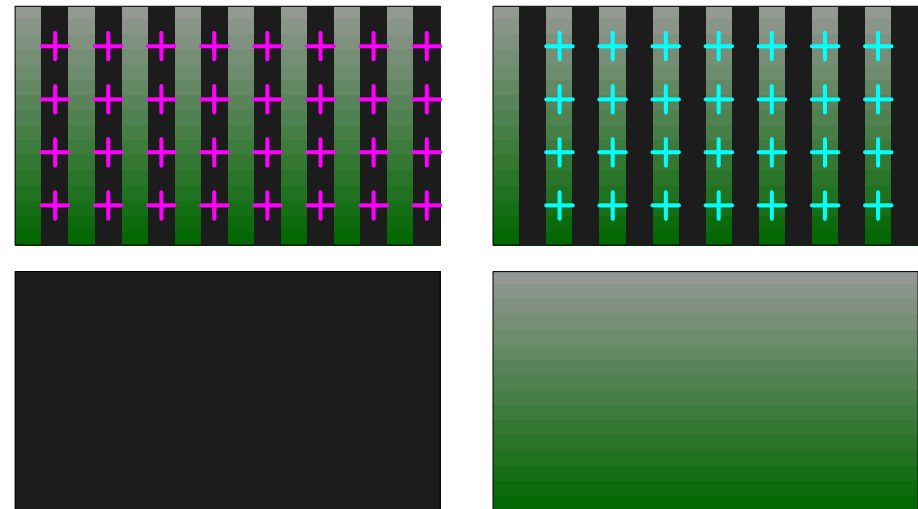
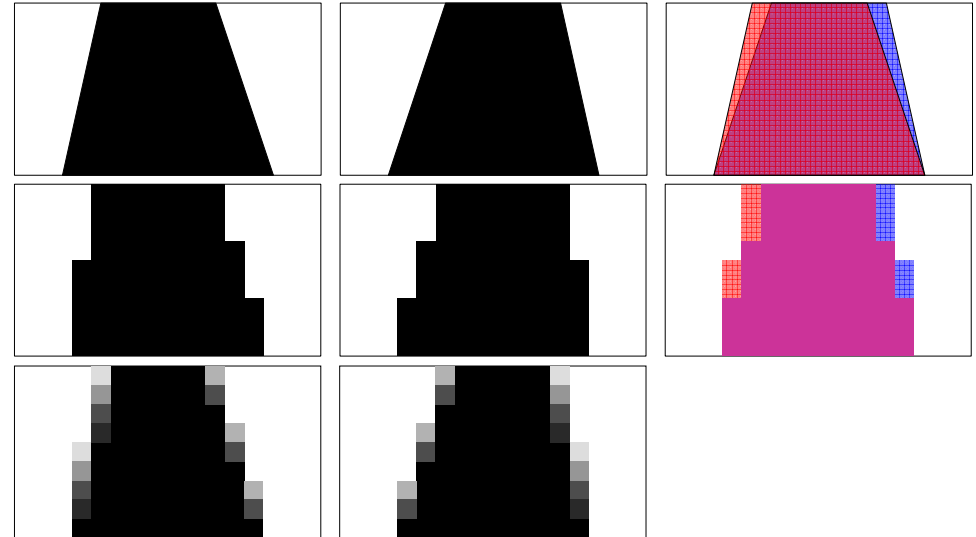
平面視で運動視差
が出たときに
気になる現象は、
立体視では動かさ
なくても気になる。

4-(1)

「動いて気になるものは、 止まっても気になる」



- サンプルング誤差
 - 水平・垂直に近いエッジのエイリアシング
 - 細い線やポリゴンが消えてしまう／テクスチャ(網目など)のツブレ・消滅
 - モアレの発生
- 描画打ち切りの影響
 - ポリゴン枚数や描画時間で打ち切ると...



- 平面視のテクニックが通用する
 - 検出方法
 - 「両眼視差は、運動視差で補完できる。」(ことが多い)
 - 「動いて気になる…」を利用
 - 動かしてみて、ちらちらしないか？
 - 平面視と同等の対策が有効
 - エッジのちらつき→アンチエイリアシング
 - サンプリング誤差、モアレ→ミップマップ
 - 描画打ち切り→「左右で違いが出ないもの」で打ち切ること
- 高フレームレートを保つ
- 平面視より目立つので、判断基準はやや厳しめに

(2) 見落としがちなこと

- **【必須】項目の落とし穴**
 - 左右を間違えない
 - デモのときには実際よく間違える...
 - 垂直視差の排除
 - 視錐台の設定が正しくできていれば...
 - 「ライブラリで対応しているから間違えようがない」
→ プレンダリングムービー制作で間違うことも...
 - 両眼間隔と画面サイズ
 - イベント会場で、巨大画面で表示！
→ 「小画面でチェックしてOK！」と思っていると大失敗
 - プレンダリングムービーは描き直せない

(2) 見落としがちなこと (つづき)



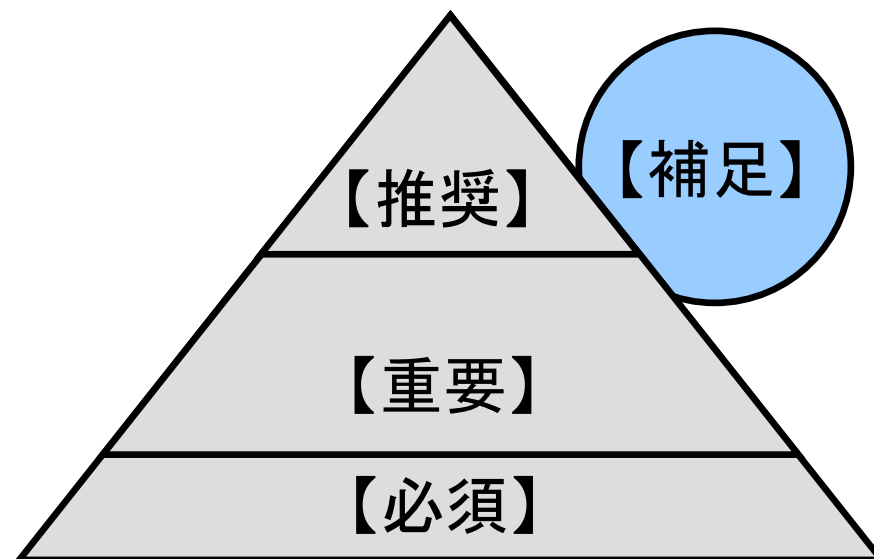
- 【重要】項目の落とし穴
 - 「飛び出しすぎ」のオブジェクトについて、
「一瞬で通り過ぎるようにした」から大丈夫！
 - →ポーズをかけたらその状態で止まる。
 - 「奥行き飛び」を解消し
スムーズにカットをつなげた！
 - →ダイジェスト版で奥行き飛びが頻発。
(映画の予告編でよく見られた。)

(3) 神経質になり過ぎない

- 映画とゲームの違い
 - 映画は、すべてのシーンを美しい配置にできる。
一度通して観ればチェックできる。
 - ゲームでは、発生しうる画面構成を
すべてチェックすることは、事実上不可能。
- 「見づらい」の発生頻度を0に近づけること
 - 「めったに起こらない」は「見づらさ」対策になりうる。
(ただし持続しないこと)
 - 普通にプレイしていれば起こらない

5. 補足とまとめ

- (1) 個人差の問題
- (2) 全ての対策を投入できないものか？
- (3) 最後に



(1)「個人差」の問題

「立体視の見え方には個人差があります。」

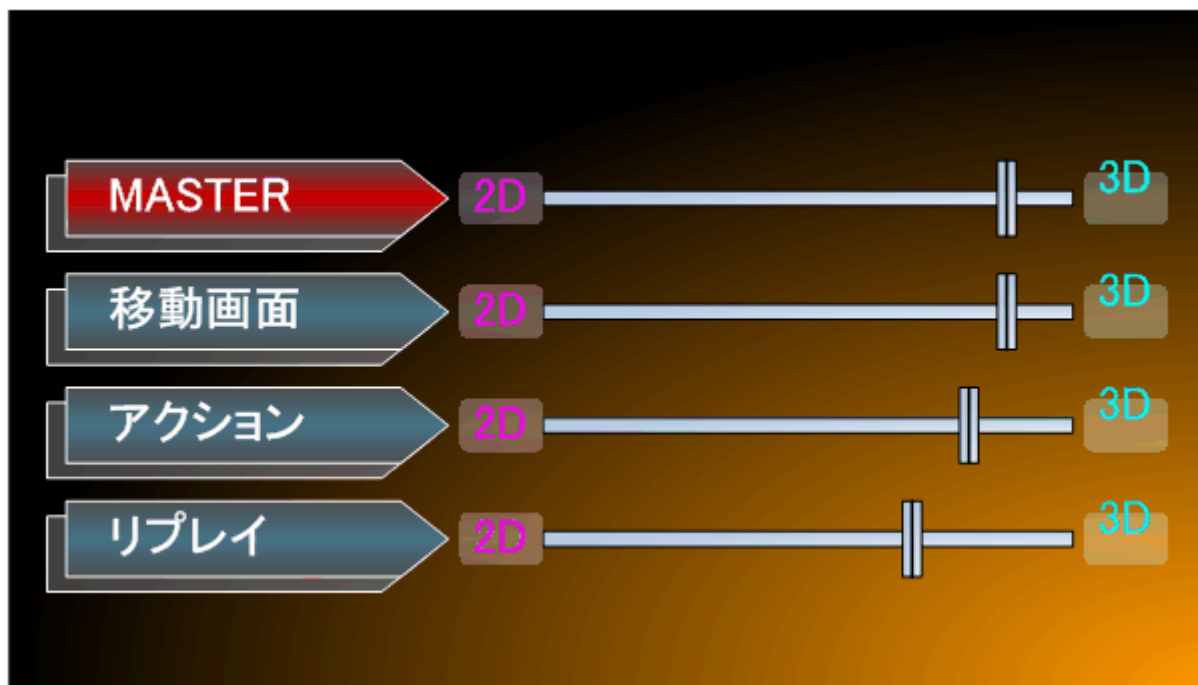
- － それは確かだが...
- － 予想以上に「個人差」は大きい。
 - →キメ細やかな個人差対応
- － 「個人差」以外にも原因が？
- － 個人差は「視覚」以外のところにも？

「視覚の」「個人差」以外にも原因が？



- 「見やすい位置」の説明が不十分なのでは？
 - 「取扱説明書」には書いてあるが...
 - 画面にどんどん近づいていってしまう傾向(本能？)
 - さりげなく「見やすい位置に誘導する」仕組みを。
- 現在のゲームは、パート(メニュー、移動、戦略、戦闘、ムービー、ミニゲーム、...)が別れているものが多い。
- 「プレイスタイル」の個人差も。
 - 「リプレイ画面だけ立体視で見ている。」

- 例えば、パートごとの立体感の設定を可能に。



(2)全ての対策を投入できないものか？



- プレイヤーとして遊んでみて、あらためて気づくことも多い。
- 納期との戦い:これはゲームに限らず...
- **1/60秒の戦い:これぞゲーム業界**
 - 「リアルタイム」ゲームは、フレームレートが重要
 - かつてのアーケードゲームは、解像度は低くても60fpsであった。
 - 現在のゲームは、「画像の情報量」の方がフレームレートより優先されることもある。

- 何が、そのゲームの「売り」か？

スピード感？

フレームレート？

疾走感・高速飛行感？

高画質？

解像感？（解像度？

アンチエイリアシング？ ...）

独特の色使い？ 芸術性？

ゲーム性！

（ゲームによってさまざま）

空間把握の手がかり？

情報の整理？

情報量？

オブジェクトの数？

テクスチャの描き込み？

立体感？

遠近感？ 空気遠近？ ...

シェーディング&ライティング？

光沢感？ ...

1/60秒の戦い:「効果」と「描画コスト」



- 「効果」と「描画コスト」のバランス
 - そのゲームの「売り」が優先される。
 - 1/60秒の取り合い
 - フレームレートを60fps→30fpsにすれば、他の表現に使える時間が2倍
 - 立体視をあきらめれば、他の表現に使える時間が2倍?
- 提言:
 - 「処理を簡略化して高速描画を実現する技術」を、立体視向けにカスタマイズする等、「使いこなし」が進めば、できることも広がる。
 - シンプルなゲームや、静止画、プレレンダーリングムービー等は、可能性を試すチャンス。

(3) 最後に

- 本日お話したことは、「**ヒント**」です。
 - **こうやれば絶対大丈夫**、というものではありません。
- **試行錯誤**はやはり必要です。
 - **細やかな気配り**による調整は見やすさに必ず貢献します。
 - どんな技法でも、「**効果的に使える範囲**」を検討してください。
- 今回の講演内容を参考にして、**効率的な調整**に役立てていただければ幸いです。
- 製品となったゲームや映画から学べるところは多いです。
 - 情報収集を行い、次期製品に役立てましょう。
 - **負荷の少ない実装方法**を開発することで、今回述べた技法をできる限り投入した、「**見やすくて迫力がある立体視ゲーム**」を！

S-3D!

質疑応答



S-3D!

※その“3D”ゲームには、「見えない S-」がついていますか？

ご清聴ありがとうございました。